The theory of successor extended by several predicates

Severine Fratani

The theory of successor extended by several predicates

Severine Fratani

Laboratoire d'Informatique Fondamentale de Marseille (LIF) CNRS : UMR6166
Université de la Méditerranée - Aix-Marseille II
Université de Provence - Aix-Marseille I

Abstract. We present a method to define unary relations $P_1, \ldots, P_n$ such that the Monadic Second-Order theory of the natural integers endowed with the successor relation and $P_1, \ldots, P_n$ is decidable. The main tool is a novel class of iterated pushdown automata whose transitions are controlled by tests on the store.

Introduction

In [7], Elgot and Rabin devise a method allowing to construct unary predicates P such that the Monadic Second-Order theory of $\langle \mathbb{N}, +1, P \rangle$ is decidable (here $+1$ denotes the successor relation). Further results in this direction have been established in [22,21,18,3,13]. This kind of problem takes place in the more general perspective of studying "weak" arithmetical theories, which possess interesting decidability properties (see [2]).

We present here a method allowing to define sequences of relations $P_1, \ldots, P_n$, such that the MSO-theory of $\langle \mathbb{N}, +1, P_1, \ldots, P_n \rangle$ is decidable. To our knowledge the only one result dealing with several relations have been given in [16] in the special case where $P_i = \{m^{2^i}\}_{m \in \mathbb{N}}$. The work here presented extends the one we made in [13], where we prove the decidability of the MSO-theory of $\langle \mathbb{N}, +1, P \rangle$ for a large class of relations P. The method consisted of consider integer sequences computed by iterated pushdown automata. These automata have been introduced in [1] as a generalization of pushdown automata and have been more studied, see e.g. [19,5,8,9,10,6], or more recently [4,17,12].

We obtain here more powerful results by the same method but by using a novel class of automata. The new feature of the automata here considered is that transitions are "controlled" by some predicates. These automata are introduced in [12,11] where conditions on controllers are given to ensure the decidability of the MSO-theory of their computation graphs. This allows to obtain two main improvements: first, results of [13] are extended to several relations, and second, these relations belong to a largest class. In particular, in [13], every relations are included in the one studied in [3], and are then "Residually Ultimately Periodic". Here we go out this class by showing, e.g., that structures $\langle \mathbb{N}, +1, n\lfloor\sqrt{n}\rfloor \rangle$ and $\langle \mathbb{N}, +1, n\lfloor log(n) \rfloor \rangle$ have a decidable MSO-theory.

1 Preliminaries

1.1 Some notations

Given a finite set A, we denote by $|A|$ the cardinal of A and by $\mathcal{P}(A)$ the powerset of A. The set of all positive integer is $\mathbb{N}$ and $\mathbb{N}^+ = \mathbb{N} - \{0\}$.

If s is a map from a set A, then $s(A) = \{s(a) \mid a \in A\}$.

1.2 Words and languages

If A is a set, A^* denotes the set of words (finite sequences) over A, ε is the empty word and $A^+ = A^* - \{\varepsilon\}$. For a given word $u \in A^*$, we denote by $|u|$ the length of u.

For $n \geq 0$ we define $A^n = \{u \in A^*.|u| = n\}$ and $A^{(n)} = \{u \in A^*.|u| \leq n\}$.

1.3 Iterated Pushdown stores

Originally defined by Greibach in [14], iterated pushdown stores are storage structures built iteratively. Let us fix an infinite sequence $\mathcal{A} = A_1, A_2, \ldots, A_k, \ldots$ of alphabets. For all $k \geq 1$, we denote by $\mathcal{A}_k$ the finite sequence $A_1, \ldots, A_k$ and adopt the convention that $\mathcal{A}_0 = \emptyset$.

Definition 1. *For $k \geq 0$, the set k-$\mathsf{pds}(\mathcal{A}_k)$ of all k-iterated pushdown stores over $\mathcal{A}_k$ is defined inductively by:*

$$0\text{-}\mathsf{pds}(\mathcal{A}_0) = \{\varepsilon\} \quad \textit{and for } k \geq 0,\ (k+1)\text{-}\mathsf{pds}(\mathcal{A}_{k+1}) = (A_{k+1}[k\text{-}\mathsf{pds}(\mathcal{A}_k)])^*.$$

The set of all iterated pushdown stores is it-$\mathsf{pds}(\mathcal{A}) = \bigcup_{k \geq 0} k\text{-}\mathsf{pds}(\mathcal{A}_k)$.

Then, every non empty ω in $(k+1)$-$\mathsf{pds}(\mathcal{A}_{k+1})$, (for $k \geq 0$), has a unique decomposition as $\omega = a[\omega_1]\omega'$ with $\omega_1 \in k$-$\mathsf{pds}(\mathcal{A}_k)$, $\omega' \in (k+1)$-$\mathsf{pds}(\mathcal{A}_{k+1})$ and $a \in A_{k+1}$. In the rest of the paper, we will often replace by a every occurrence of $a[\varepsilon]$ appearing in the description of a k-pds.

Example 1. Let $A_1 = \{a_1, b_1\}$, $A_2 = \{a_2, b_2\}$ and $A_3 = \{a_3, b_3\}$ be alphabets, and $\omega_{ex} = b_3[b_2[b_1[\varepsilon]a_1[\varepsilon]]a_2[a_1[\varepsilon]]]a_3[\varepsilon]a_3[a_2[a_1[\varepsilon]b_1[\varepsilon]]] \in 3\text{-}\mathsf{pds}(\mathcal{A}_3)$. It can be written $\omega_{ex} = b_3[b_2[b_1a_1]a_2[a_1]]a_3a_3[a_2[a_1b_1]]$, and its decomposition is $\omega_{ex} = a[\omega_1]\omega'$ with $a = b_3$, $\omega_1 = b_2[b_1a_1]a_2[a_1]$ and $\omega' = a_3a_3[a_2[a_1b_1]]$.

The two following maps will be useful.

Projection: the map associating any it-pds to its top i-pds, $1 \leq i$ is
p_i: it-$\mathsf{pds}(\mathcal{A}) \to i$-$\mathsf{pds}(\mathcal{A}_i)$, defined for all $\omega \in k$-$\mathsf{pds}(\mathcal{A}_k)$ by:
if $k < i$ then $\mathrm{p}_i(\omega)$ is undefined,
if $k = i$ then $\mathrm{p}_i(\omega) = \omega$,
if $k > i$ then $\mathrm{p}_i(\omega) = \mathrm{p}_i(\omega_1)$ if $\omega = a[\omega_1]\omega'$ and $\mathrm{p}_i(\omega) = \varepsilon$ if $\omega = \varepsilon$.

Top symbols: the map associating any it-pds to its top symbols is top : it-$\mathsf{pds}(\mathcal{A}) \to \mathcal{A}^*$ defined by:

$$\mathrm{top}(\varepsilon) = \varepsilon \text{ and } \mathrm{top}(a[\omega_1]\omega') = a \cdot \mathrm{top}(\omega_1).$$

Let $i \in [1, k]$, and $\omega \in k\text{-}\mathsf{pds}$, if $|\mathrm{top}(\omega)| \geq i$, then $\mathrm{top}_i(\omega)$ is the i-th letter of $\mathrm{top}(\omega)$, else $\mathrm{top}_i(\omega) = \varepsilon$.

Example 2. Let ω_{ex} be the 3-pds given in Example 1:
$\mathrm{p}_2(\omega_{ex}) = b_2[b_1a_1]a_2[a_1]$, $\mathrm{p}_1(\omega_{ex}) = b_1a_1$, and
$\mathrm{top}(\omega_{ex}) = b_3b_2b_1$, $\mathrm{top}(\mathrm{p}_2(\omega_{ex})) = b_2b_1$, $\mathrm{top}(\mathrm{p}_1(\omega_{ex})) = b_1$.

A **pushdown instruction** is a map from $it\text{-}\mathsf{pds}(\mathcal{A})$ to $it\text{-}\mathsf{pds}(\mathcal{A})$ which does not modify the level of the pushdowns (i.e., if instr is an instruction, then for any $k \geq 1$ and any $\omega \in k\text{-}\mathsf{pds}$, $\mathsf{instr}(\omega) \in k\text{-}\mathsf{pds}$). An **instruction of level** i is an instruction which does not modify the levels greater than i of any it-pds. Hence, given instr an instruction of level i and $\omega = a[\omega_1]\omega' \in k\text{-}\mathsf{pds}$:

if $k > i$, then $\mathsf{instr}(\omega) = a[\mathsf{instr}(\omega_1)]\omega'$ and $\mathsf{instr}(\varepsilon) = \varepsilon$,

if $k < i$, then $\mathsf{instr}(\omega) = \omega$ and $\mathsf{instr}(\varepsilon) = \varepsilon$.

Therefore, to define an instruction of level i, we just need to define it for any stack $\omega \in i\text{-}\mathsf{pds}(\mathcal{A}_i)$.
Four instructions are generally applicable to it-pushdowns.

Definition 2. *For any $i \geq 1$, "classical" instructions of level i over $\mathcal{A}$ are defined by: for all $\omega = a[\omega_1]\omega' \in i\text{-}\mathsf{pds}(\mathcal{A}_i)$, for all $b \in A_i$,*

- $\mathrm{pop}_i(\omega) = \omega'$ *and* $\mathrm{pop}_i(\varepsilon)$ *is undefined,*
- $\mathrm{push}_b(\omega) = b[\omega_1]\omega$ *and* $\mathrm{push}_b(\varepsilon) = b$*,*
- $\mathrm{change}_b(\omega) = b[\omega_1]\omega'$ *and* $\mathrm{change}_b(\varepsilon)$ *is undefined,*
- $\mathrm{stay}(\omega) = \omega$ *and* $\mathrm{stay}(\varepsilon) = \varepsilon$.

For $k \geq 1$, $\mathcal{I}_k(\mathcal{A}_k) = \{stay\} \cup \{\mathrm{pop}_i\}_{i \in [1,k]} \cup \{\mathrm{push}_a, \mathrm{change}_a\}_{a \in \mathcal{A}_k}$ is the set of instructions over $\mathcal{A}_k$.

Then, given $\omega \in k\text{-}\mathsf{pds}(\mathcal{A}_k)$, $i \in [1, k]$ and $b \in A_i$, $\mathrm{pop}_i(\omega)$ erases $\mathrm{p}_i(\omega)$ on the top of the store, $\mathrm{push}_b(\omega)$ consists in add $b[\mathrm{p}_{i-1}(\omega)]$ on the top of the top i-pds and $\mathrm{change}_b(\omega)$ consists in replace $\mathrm{top}_i(\omega)$ by b.

Example 3. Let $\omega = b_3[b_2[b_1a_1]a_2[a_1]]a_3[b_2]$ be a 3-pds:
$\mathrm{pop}_3(\omega) = a_3[b_2]$, $\mathrm{pop}_2(\omega) = b_3[a_2[a_1]]a_3[b_2]$,
$\mathrm{pop}_1(\omega) = b_3[b_2[a_1]a_2[a_1]]a_3[b_2]$,
$\mathrm{push}_{a_3}(\omega) = a_3[b_2[b_1a_1]a_2[a_1]]b_3[b_2[b_1a_1]a_2[a_1]]a_3[b_2]$,
$\mathrm{push}_{a_2}(\omega) = b_3[a_2[b_1a_1]b_2[b_1a_1]a_2[a_1]]a_3[b_2]$,
$\mathrm{push}_{a_1}(\omega) = b_3[b_2[a_1b_1a_1]a_2[a_1]]a_3[b_2]$,
$\mathrm{change}_{a_3}(\omega) = a_3[b_2[b_1a_1]a_2[a_1]]a_3[b_2]$,
$\mathrm{change}_{a_2}(\omega) = b_3[a_2[b_1a_1]a_2[a_1]]a_3[b_2]$,
$\mathrm{change}_{a_1}(\omega) = b_3[b_2[a_1a_1]a_2[a_1]]a_3[b_2]$.

1.4 Iterated Pushdown Automata and extensions.

We define here iterated pushdown automata (it-pda) and a particular class of controlled iterated pushdown automata. We suppose fixed an infinite sequence $\mathcal{A} = A_1, \ldots, A_k, \ldots$ of stack alphabets.

Definition 3 (Iterated pushdown automata). *Let $k \geq 1$, a k-pda is a structure $\mathcal{A} = (Q, \Sigma, \mathcal{A}_k, \Delta, q_0, Z)$ where Q is a finite set of states, Σ is a terminal alphabet, $q_0 \in Q$ is the initial state, $Z \in A_k$ is the initial symbol, and $\Delta \subseteq Q \times \Sigma \times {\mathcal{A}_k}^{(k)} - \{\varepsilon\} \times \mathcal{I}_k(\mathcal{A}_k) \times Q$ is the transition relation.*
The family of all k-pda over the stack alphabets $\mathcal{A}_k$ is k-$\mathsf{PDA}(\mathcal{A}_k)$ (or k-PDA when $\mathcal{A}_k$ is understood). The set of configurations *of $\mathcal{A}$ is $Con_{\mathcal{A}} = Q \times \Sigma^* \times k\text{-}\mathsf{pds}(\mathcal{A}_k)$. The single step relation $\rightarrow_{\mathcal{A}} \subseteq Con_{\mathcal{A}} \times Con_{\mathcal{A}}$ of $\mathcal{A}$ is defined by*

$$(p, \alpha\sigma, \omega) \rightarrow_{\mathcal{A}} (q, \sigma, \omega') \textit{ iff } (p, \alpha, \mathrm{top}(\omega), instr, q) \in \Delta, \textit{ and } \omega' = instr(\omega).$$

We denote by $\overset{*}{\rightarrow}_{\mathcal{A}}$ the reflexive and transitive closure of $\rightarrow_{\mathcal{A}}$. The language recognized by *$\mathcal{A}$ is $\mathrm{L}(\mathcal{A}) = \{\sigma \in \Sigma^* \mid \exists q \in F, (q_0, \sigma, Z) \overset{*}{\rightarrow}_{\mathcal{A}} (q, \varepsilon, \varepsilon)\}$.*

Counter pushdown automata are 1-pda whose stack alphabet is reduced to a unique letter. The stack can then be seen as an integer. We extend this notion to *it*-pda: a counter *it*-pda is an *it*-pda whose stack alphabet of level 1 (i.e., A_1) is reduced to a single letter. Now we define controlled counter *it*-pda (*it*-cpda) which are counter *it*-pda whose transitions are controlled by tests on the top counter of the stack. Initially, controlled *it*-pda have been introduced in [11,12].

Definition 4 (Controlled counter iterated pushdown automata). *Let $k \geq 0$, a k-cpda is a structure $\mathcal{A} = (Q, \Sigma, \mathcal{A}_k, \boldsymbol{N}, \Delta, q_0, Z)$ where Q, Σ, q_0 and Z are defined as previously, , $\mathcal{A}_k = A_1, \ldots, A_k$, with $|A_1| = 1$, $\boldsymbol{N} = (N_1, \ldots, N_m)$ is a vector of subsets of $\mathbb{N}$ called* controllers *and $\Delta \subseteq Q \times \Sigma \times {\mathcal{A}_k}^{(k)} - \{\varepsilon\} \times \{0,1\}^m \times \mathcal{I}_k(\mathcal{A}_k) \times Q$ is the transition relation.*
The family of all k-cpda controlled by $\boldsymbol{N}$, over the pushdown alphabets $\mathcal{A}_k$ is k-$\mathsf{CPDA}(\mathcal{A}_k)^{\boldsymbol{N}}$ (or k-$\mathsf{CPDA}^{\boldsymbol{N}}$ when $\mathcal{A}_k$ is understood). The set of configurations *of $\mathcal{A}$ is $Con_{\mathcal{A}} = Q \times \Sigma^* \times k\text{-}\mathsf{pds}(\mathcal{A}_k)$. The single step relation $\rightarrow_{\mathcal{A}} \subseteq Con_{\mathcal{A}} \times Con_{\mathcal{A}}$ of $\mathcal{A}$ is defined by*

$$(p, \alpha\sigma, \omega) \rightarrow_{\mathcal{A}} (q, \sigma, \omega') \textit{ iff } (p, \alpha, \mathrm{top}(\omega), \chi_{\boldsymbol{N}}(|\mathrm{p}_1(\omega)|), instr, q) \in \Delta, \textit{ and } \omega' = instr(\omega),$$

where for all $n \geq 0$, $\chi_{\boldsymbol{N}}(n)$ is the boolean vector $(o_1, \ldots, o_m)$ fulfilling [$o_i = 1$ iff $n \in N_i$], $\forall i \in [1, m]$. The relation $\overset{*}{\rightarrow}_{\mathcal{A}}$ and the language recognized by $\mathcal{A}$ are defined as previously.

Remark that an automaton in k-$\mathsf{CPDA}^{\emptyset}$ can be seen as a counter k-pda, without controllers. We denote by k-CPDA the class of such automata, and we omit the test vector $\boldsymbol{o}$ in the description of their transitions.

Sometimes, we will write the transition relation Δ of an automata in k-$\mathsf{CPDA}^{\boldsymbol{N}}$ as a map $\Delta : Q \times \Sigma \times {\mathcal{A}_k}^{(k)} - \{\varepsilon\} \times \{0,1\}^m \rightarrow \mathcal{P}(\mathcal{I}_k(\mathcal{A}_k) \times Q)$.

Example 4. Let $\mathcal{A}_2 = (\{a_1\}, \{a_2, b_2\})$, and $N \subseteq \mathbb{N}$. The following automaton $\mathcal{A} \in 2\text{-}\mathsf{CPDA}^N(\mathcal{A}_2)$ fulfills : $\mathrm{L}(\mathcal{A}) = \{\alpha^n\beta^n\gamma^n \mid n \in N\}$.

$\mathcal{A} = (\{q_0, q_1\}, \{\alpha, \beta, \gamma\}, \mathcal{A}_2, N, \Delta, q_0, a_2)$ with:
$\Delta(q_0, \varepsilon, a_2, 1) = \{(\mathrm{pop}_2, q_0)\}$,
$\Delta(q_0, \alpha, a_2, o) = \Delta(q_0, \alpha, a_2 a_1, o) = \{(\mathrm{push}_{a_1}, q_0)\}$, for all $o = 0, 1$,

$\Delta(q_0, \varepsilon, a_2a_1, 1) = \{(\text{push}_{b_2}, q_1)\}$,
$\Delta(q_1, \beta, b_2a_1, o) = \Delta(q_1, \gamma, a_2a_1, o) = \{(\text{pop}_1, q_1)\}$, for all $o = 0, 1$,
$\Delta(q_1, \varepsilon, b_2, o) = \Delta(q_1, \varepsilon, a_2, o) = \{(\text{pop}_2, q_1)\}$, for all $o = 0, 1$.

Suppose that N is the set of all prime numbers, here is a computation of the word $\alpha^2\beta^2\gamma^2$:
$(q_0, \alpha^2\beta^2\gamma^2, a_2[\varepsilon]) \to (q_0, \alpha\beta^2\gamma^2, a_2[a_1]) \to (q_0, \beta^2\gamma^2, a_2[a_1a_1]) \to$ (since $2 \in N$) $(q_1, \beta^2\gamma^2, b_2[a_1a_1]a_2[a_1a_1]) \to (q_1, \beta\gamma^2, b_2[a_1]a_2[a_1a_1]) \to (q_1, \gamma^2, b_2a_2[a_1a_1]) \to (q_1, \gamma^2, a_2[a_1a_1]) \to (q_1, \gamma, a_2[a_1]) \to (q_1, \varepsilon, a_2) \to (q_1, \varepsilon, \varepsilon)$.

1.5 Deterministic automata

Two transitions $(p, \alpha, w, \boldsymbol{o}, instr, q)$ and $(p', \alpha', w', \boldsymbol{o'}, instr', q')$ of a k-cpda are said to be **compatible** iff $p = p'$, $w = w'$, $\boldsymbol{o} = \boldsymbol{o'}$ and

$$[\alpha \neq \varepsilon \text{ and } \alpha = \alpha'] \text{ or } [\alpha = \varepsilon] \text{ or } [\alpha' = \varepsilon].$$

A k-cpda is deterministic iff for every transitions $\delta, \delta' \in \Delta$, $\delta = \delta'$ or δ and δ' are incompatible. The class of all deterministic automata in k-$\mathsf{CPDA}^{\boldsymbol{N}}$ is k-$\mathsf{DCPDA}^{\boldsymbol{N}}$.

For a deterministic automaton, we will often write Δ as a map: $\Delta : Q \times \Sigma \times {\mathcal{A}_k}^{(k)} - \{\varepsilon\} \times \{0,1\}^m \to \mathcal{I}_k(\mathcal{A}_k) \times Q$.

1.6 Monadic Second-Order Logic

Let $Var = \{x, y, z, \ldots, X, Y, Z \ldots\}$ be a set of variables where $x, y, \ldots$ denote first order variables and $X, Y, \ldots$ second order variables and Sig be a signature. The set MSO(Sig) of MSO-formulas over Sig is the smallest set such that:

- $x \in X$ and $Y \subseteq X$ are MSO-formulas for every $x, Y, X \in Var$
- $r(x_1, \ldots x_\rho)$ is an MSO-formula for every $r \in Sig$, of arity ρ and every first order variables $x_1, \ldots x_\rho \in Var$
- if ϕ, ψ are MSO-formulas then $\neg\phi$, $\phi \vee \psi$, $\exists x.\phi$ and $\exists X.\phi$ are MSO-formulas.

Let $\mathcal{S} = \langle D_{\mathcal{S}}, r_1, \ldots, r_n \rangle$ be a structure over the signature Sig, a valuation of Var over $D_{\mathcal{S}}$ is a function $val : Var \to D_{\mathcal{S}} \cup \mathcal{P}(D_{\mathcal{S}})$ such that for every $x, X \in Var$, $val(x) \in D_{\mathcal{S}}$ and $val(X) \subseteq D_{\mathcal{S}}$.
The satisfiability of an MSO-formula in the structure $\mathcal{S}$ with valuation val is then defined by induction on the structure of the formula, in the usual way.
An MSO-formula $\phi(\bar{x}, \bar{X})$ (where $\bar{x} = (x_1, \ldots, x_\rho)$ and $\bar{X} = (X_1, \ldots, X_\tau)$ denote free first and second order variables of ϕ) over Sig is said to be **satisfiable in** $\mathcal{S}$ if there exists a valuation val such that $\mathcal{S}, val \models \phi(\bar{x}, \bar{X})$.
We will often abbreviate $\mathcal{S}, [\bar{x} \mapsto \bar{a}, \bar{X} \mapsto \bar{A}] \models \phi(\bar{x}, \bar{X})$ by $\mathcal{S} \models \phi(\bar{a}, \bar{A})$.

Definition 5. *A structure $\mathcal{S}$ admits a decidable MSO-theory if for every MSO-sentence ϕ (i.e. MSO-formula without free variables) one can effectively decide whether $\mathcal{S} \models \phi$.*

A subset D of $D_{\mathcal{S}}$ is said to be **MSO-definable** in $\mathcal{S}$ iff there exists $\phi(X)$ in MSO(Sig) such that:

$$\mathcal{S} \models \phi(D) \text{ and } \forall S \subseteq D_{\mathcal{S}}, \text{ if } \mathcal{S} \models \phi(S) \text{ then } S = D\text{s}.$$

$Sig = \{r_1, \ldots, r_n\}$ (resp. $Sig' = \{r'_1, \ldots, r'_m\}$) be some relational signature and $\mathcal{S}$ (resp. $\mathcal{S}'$) be some structure over the signature Sig (resp. Sig').

Definition 6 (Interpretations). *An* MSO-interpretation *of the structure* $\mathcal{S}$ *into the structure* $\mathcal{S}'$ *is an injective map* $f : D_{\mathcal{S}} \to D_{\mathcal{S}'}$ *such that,*

1. $f(D_{\mathcal{S}})$ *is* MSO-*definable in* $\mathcal{S}'$
2. $\forall i \in [1, n]$*, there exists* $\phi'_i(\bar{x}) \in MSO(Sig')$*, (where* $\bar{x} = x_1, \ldots, x_{\rho_i}$*) fulfilling that, for every valuation* val *of* Var *in* $D_{\mathcal{S}}$

$$(\mathcal{S}, val) \models r_i(\bar{x}) \Leftrightarrow (\mathcal{S}', f \circ val) \models \phi'_i(\bar{x}).$$

Theorem 1 ([20]). *Suppose there exists a computable* MSO**-interpretation** *of the structure* $\mathcal{S}$ *into the structure* $\mathcal{S}'$*. If* $\mathcal{S}'$ *has a decidable* MSO*-theory, then* $\mathcal{S}$ *has a decidable* MSO*-theory too.*

1.7 Logic over iterated-pushdowns

Let $\mathcal{A}$ be a sequence of alphabets, computations of an automaton in k-$\mathsf{PDA}(\mathcal{A}_k)$ are naturally expressed by MSO formulas in the following structure:

$$\mathsf{PDS}_k(\mathcal{A}_k) = \langle k\text{-}\mathsf{pds}(\mathcal{A}_k), (\text{TOP}_u)_{u \in \mathcal{A}_k{}^{(k)}}, (\text{POP}_i, \text{PUSH}_a, \text{CHANGE}_a,)_{i \in [1,k], a \in \mathcal{A}_k} \rangle.$$

Relations POP_i, PUSH_a, CHANGE_a and TOP_u are graphs of the corresponding instructions on pushdowns.

Theorem 2 ([13, Theorems 30 and 32]). *The MSO-theory of* $\mathsf{PDS}_k(\mathcal{A}_k)$ *is decidable, for all* $k \geq 1$.

Computations of an automaton in k-$\mathsf{CPDA}(\mathcal{A}_k)^{\boldsymbol{N}}$, with $\boldsymbol{N} = (N_1, \ldots, N_m)$, are expressed in the extended structure $\mathsf{PDS}_k(\mathcal{A}_k)^{\boldsymbol{N}}$ obtained from $\mathsf{PDS}_k(\mathcal{A}_k)$ by adding the unary relations $pN_1, \ldots, pN_n$ where $pN_i = \{\omega \in k\text{-}\mathsf{pds}(\mathcal{A}_k),\ |\mathrm{p}_1(\omega)| \in N_i\}$.

Theorem 3 ([12, Theorem 6.2.2],[11]). *If* $\boldsymbol{N}$ *is a vector of subsets of* $\mathbb{N}$*, and the MSO-theory of* $\langle \mathbb{N}, +1, N_1, \ldots, N_m \rangle$ *is decidable, then the MSO-theory of* $\mathsf{PDS}_k(\mathcal{A}_k)^{\boldsymbol{N}}$ *is decidable.*

1.8 Sequences

A *sequence* of natural numbers is any map $u : \mathbb{N} \to \mathbb{N}$. Such a sequence u can be also viewed as a formal power series

$$u(X) = \sum_{n=0}^{\infty} u_n X^n.$$

The following operators on series are classical:
E: the *shift* operator: $(\mathsf{E}u)(n) = u(n+1)$; $\quad (\mathsf{E}u)(X) = \frac{u(X)-u(0)}{X}$
Δ: the difference operator

$$(\Delta u)(n) = u(n+1) - u(n); \qquad (\Delta u)(X) = \frac{u(X)(1-X) - u(0)}{X}$$

Σ: the summation operator $(\Sigma u)(n) = \sum_{j=0}^{n} u(j)$; $\quad (\Sigma u)(X) = \frac{u(X)}{1-X}$
$+$: the sum operator

$$(u+v)(n) = u(n) + v(n); \qquad (u+v)(X) = u(X) + v(X)$$

$\cdot$: the external product, for every $r \in \mathbb{Q}$ $(r \cdot u)(n) = r \cdot u(n)$
$\odot$: the Hadamard product, (also called the “ordinary“ product)

$$(u \odot v)(n) = u(n) \cdot v(n)$$

$\times$: the convolution product

$$(u \times v)(n) = \sum_{k=0}^{n} u(k) \cdot v(n-k); \qquad (u \times v)(X) = u(X) \cdot v(X)$$

$^{-1}$: the operator ”inverse”, for u strictly increasing,

$$u^{-1}(n) = |u(\mathbb{N}^+) \cap [0, \ldots, n]|$$

$\circ$: the sequence composition $(u \circ v)(n) = u(v(n))$
$\bullet$: the series composition : if $v(0) = 0$, $(u \bullet v)(X) = \sum_{n=0}^{\infty} u(n) \cdot v(X)^n$.

2 Sequences defined by automata

We define here a class of *integer sequences* by means of k-cpda. We show that the class of sequences thus defined contains numerous classes of recursive sequences and is closed under many natural operations.

Definition 7 ($(k, \boldsymbol{N})$-computable sequences). *Let $\boldsymbol{N}$ be a vector of subsets of $\mathbb{N}$. A sequence of natural integers s is called a $(k, \boldsymbol{N})$*-computable sequence *iff there exists $\mathcal{A} \in k\text{-}\mathsf{DCPDA}(\mathcal{A}_k)^{\boldsymbol{N}}$, defined over the pushdown alphabets $\mathcal{A}_k = A_1, \ldots, A_k$ where each A_i contains a letter a_i, and such that for all $n \geq 0$:*

$$(q_0, \alpha^{s(n)}, a_1[a_2 \ldots [a_{k-1}[{a_k}^n]] \ldots]) \xrightarrow{*}_{\mathcal{A}} (q_0, \varepsilon, \varepsilon).$$

We denote by $\mathbb{S}_k^{\boldsymbol{N}}$ the set of all $(k, \boldsymbol{N})$-computable sequences of natural integers (or $\mathbb{S}_k$ if $\boldsymbol{N} = \emptyset$).

This computation scheme is well adapted to recurrent sequences. Let us expose the principle with a simple example.

Example 5 (Linear recurrence). Let s be the sequence defined by

$$s(0) = 2;\ \ \forall n \geq 0,\ s(n+1) = 2s(n) + 1.$$

Suppose that there exists $\mathcal{A} \in$ 2-DCPDA such that:

1. $\forall \omega \in$ 2-pds, $(q_0, \alpha^{s(0)}, a_2[\varepsilon]\omega) \xrightarrow{*}_{\mathcal{A}} (q_0, \varepsilon, \omega)$,
2. $\forall n \geq 0$, $\forall \omega \in$ 2-pds, $(q_0, \varepsilon, a_2[{a_1}^{n+1}]\omega) \xrightarrow{*}_{\mathcal{A}} (q_0, \varepsilon, b_2[{a_1}^n]a_2[{a_1}^n]a_2[{a_1}^n]\omega)$,
3. $\forall n \geq 0$, $\forall \omega \in$ 2-pds, $(q_0, \alpha, b_2[{a_1}^n]\omega) \xrightarrow{*}_{\mathcal{A}} (q_0, \varepsilon, \omega)$.

Let us check by induction over $n \geq 0$ that such an automaton fulfills the following property $\mathbf{P}(n)$: $\forall \omega \in$ 2-pds,

$$(q_0, \alpha^{s(n)}, a_2[{a_1}^n]\omega) \xrightarrow{*}_{\mathcal{A}} (q_0, \varepsilon, \omega).$$

Hypothesis (1) proves $\mathbf{P}(0)$. Suppose $\mathbf{P}(n)$ for $n \geq 0$. For every $\omega \in$ 2-pds, we obtain by applying hypothesis (2), hypothesis (3), then two times $\mathbf{P}(n)$:

$$\begin{aligned}(q_0, \alpha^{s(n+1)}, a_2[{a_1}^{n+1}]\omega) &\xrightarrow{*}_{\mathcal{A}} (q_0, \alpha^{s(n+1)}, b_2[{a_1}^n]a_2[{a_1}^n]a_2[{a_1}^n]\omega)\\ &\xrightarrow{*}_{\mathcal{A}} (q_0, \alpha^{2s(n)}, a_2[{a_1}^n]a_2[{a_1}^n]\omega)\\ &\xrightarrow{*}_{\mathcal{A}} (q_0, \alpha^{s(n)}, a_2[{a_1}^n]\omega)\\ &\xrightarrow{*}_{\mathcal{A}} (q_0, \varepsilon, \omega).\end{aligned}$$

Then, $\mathbf{P}(n)$ is true for every $n \geq 0$, and in the particular case where $\omega = \varepsilon$, $\mathcal{A}$ computes the sequence s.

Let us prove that there exists a deterministic 2-pda fulfilling hypothesis (1), (2) and (3). Let $\mathcal{A} = (\{q_0, q_1, q_2\}, \{\alpha\}, \mathcal{A}_2, \Delta, q_0, a_2)$ where $A_1 = \{a_1\}$, $A_2 = \{a_2, b_2\}$ and:

(a) $\Delta(q_0, \alpha, a_2) = (\text{change}_{b_2}, q_0)$,
(b) $\Delta(q_0, \varepsilon, a_2 a_1) = (\text{pop}_1, q_1)$ and
$\Delta(q_1, \varepsilon, a_2 a_1) = \Delta(q_1, \varepsilon, a_2) = (\text{push}_{a_2}, q_2)$ and
$\Delta(q_2, \varepsilon, a_2 a_1) = \Delta(q_2, \varepsilon, a_2) = (\text{push}_{b_2}, q_0)$,
(c) $\Delta(q_0, \alpha, b_2) = \Delta(q_0, \alpha, b_2 a_1) = (\text{pop}_2, q_0)$.

This automaton is deterministic, transitions (a) and (c) allow to obtain hypothesis (1), transitions (b) makes true hypothesis (2), and transitions (c) allow the computation (3).

2.1 Some computable sequences

Definition 8 ($\mathbb{N}$-rational sequences). *A sequence $(u_n)_{n\geq 0}$ is $\mathbb{N}$-rational iff there is a matrix M in $\mathbb{N}^{d\times d}$ and two vectors L in $\mathbb{B}^{1\times d}$ and C in $\mathbb{B}^{d\times 1}$ such that $u_n = L \cdot M^n \cdot C$.*

Proposition 1 ([13, Prop. 50]). *If $(u_n)_{n\geq 0}$ is $\mathbb{N}$-rational, then $(u_n)_{n\geq 0} \in \mathbb{S}_2$.*

Proposition 2 ([13, Prop. 53]). *Let $P_i(X_1, \ldots, X_p)$, $(1 \leq i \leq p)$ be polynomials with coefficients in $\mathbb{N}$, $c_1, \ldots, c_i, \ldots c_p \in \mathbb{N}$ and , u_i $(1 \leq i \leq p)$ be the sequence defined by $u_i(n+1) = P_i(u_1(n), \ldots, u_p(n))$, and $u_i(0) = c_i$. Then $u_1 \in \mathbb{S}_3$.*

Proposition 3. *Let s be a strictly increasing sequence of natural numbers, then $s^{-1} \in \mathbb{S}_2^{s(\mathbb{N}^+)}$.*

Theorem 4.
0- For every $f \in \mathbb{S}_{k+1}^{\boldsymbol{N}}$, $k \geq 1$, and every integer $c \in \mathbb{N}$, sequences Ef and $f + \frac{c}{1-X}$, belong to $\mathbb{S}_{k+1}^{\boldsymbol{N}}$; if $\forall n \in \mathbb{N}, f(n) \geq c$ then $f - \frac{c}{1-X}$ belongs to $\mathbb{S}_{k+1}^{\boldsymbol{N}}$; the sequence $0 \mapsto c, n+1 \mapsto f(n)$ belongs to $\mathbb{S}_{k+1}^{\boldsymbol{N}}$.
1- For every $f, g \in \mathbb{S}_{k+1}^{\boldsymbol{N}}$, with $k \geq 1$, the sequence $f + g$ belongs to $\mathbb{S}_{k+1}^{\boldsymbol{N}}$.
2- For every $f, g \in \mathbb{S}_{k+1}^{\boldsymbol{N}}$, with $k \geq 2$, the sequence $f \odot g$, belongs to $\mathbb{S}_{k+1}^{\boldsymbol{N}}$ and for every $f' \in \mathbb{S}_{k+2}^{\boldsymbol{N}}$, f'^g belongs to $\mathbb{S}_{k+2}^{\boldsymbol{N}}$.
3- For $f \in {\mathbb{S}_{k+1}}^{\boldsymbol{N}}$, $g \in \mathbb{S}_k$, $k \geq 2$, sequences $f \times g$ and $f \bullet g$ belong to $\mathbb{S}_{k+1}^{\boldsymbol{N}}$.
4- For every $g \in \mathbb{S}_k$, with $k \geq 2$, the sequence f defined by: $f(n+1) = \sum_{m=0}^{n} f(m) \cdot g(n-m)$ and $f(0) = 1$ (the convolution inverse of $1 - X \times f$) belongs to $\mathbb{S}_{k+1}$.
5- For every $f \in \mathbb{S}_k$, $g \in \mathbb{S}_\ell^{\boldsymbol{N}}$, for $k, l \geq 2$, the sequence $f \circ g$ belongs to $\mathbb{S}_{k+\ell-1}^{\boldsymbol{N}}$.
6- For every $k \geq 2$ and for every system of recurrent equations expressed by polynomials in $\mathbb{S}_{k+1}^{\boldsymbol{N}}[X_1, \ldots, X_p]$, with initial conditions in $\mathbb{N}$, every solution belongs to $\mathbb{S}_{k+1}^{\boldsymbol{N}}$.
7- For every $k \geq 2$ and for every system of recurrent equations expressed by polynomials with coefficients in $\mathbb{S}_{k+2}^{\boldsymbol{N}}$, exponents in $\mathbb{S}_{k+1}^{\boldsymbol{N}}$ and initial conditions in $\mathbb{N}$, every solution belongs to $\mathbb{S}_{k+2}^{\boldsymbol{N}}$.

3 Application to the sequential calculus

We combine now the decidability theorems about k-pda structures presented in Section 1.7 and the results obtained in Section 2 to prove the decidability of the MSO-theory of structures $\langle \mathbb{N}, +1, P \rangle$, for a large class of relations P (Theorem 5 and Theorem 8) containing for example $(n\lfloor\sqrt{n}\rfloor)_{n\in\mathbb{N}}$ or $(n^2\lfloor\log n\rfloor)_{n\in\mathbb{N}}$. These results are generalized to the case of structures with several relations (Theorem 7), as for example

$$\langle \mathbb{N}, +1, \{n^{k_1}\}_{n\geq 0}, \{n^{k_1 k_2}\}_{n\geq 0}, \ldots, \{n^{k_1 \cdots k_m}\}_{n\geq 0} \rangle, \text{ for } k_1, \ldots, k_m \geq 0.$$

3.1 Extensions of $\langle \mathbb{N}, +1 \rangle$

It is proved in [13] that for every sequence s calculated (in the sense of Definition 7) by an automaton in k-$\mathsf{DCPDA}(\mathcal{A}_k)$, the structure $\langle \mathbb{N}, +1, \Sigma s(\mathbb{N}) \rangle$ is interpretable inside the structure $\mathsf{PDS}_k(\mathcal{A}_k)$, and since this structure has a decidable MSO-theory (Theorem 2), it follows from Theorem 1:

Theorem 5 ([13, Theorem 82]). *For every $s \in \mathbb{S}_k$, $k \geq 1$, the* MSO*-theory of $\langle \mathbb{N}, +1, \Sigma s(\mathbb{N})\rangle$ is decidable.*

By the same proof, we can show that for every sequence s calculated by an automaton in k-$\mathsf{DCPDA}(\mathcal{A})^{\boldsymbol{N}}$, the structure $\langle \mathbb{N}, +1, \Sigma s(\mathbb{N})\rangle$ is interpretable inside the structure $\mathsf{PDS}_k(\mathcal{A}_k)^{\boldsymbol{N}}$. Then using Theorem 3, we get:

Theorem 6. *If $s \in \mathbb{S}_k^{\boldsymbol{N}}$, with $\boldsymbol{N} = (N_1, \ldots, N_m)$ such that $\langle \mathbb{N}, +1, N_1, \ldots, N_m\rangle$ has a decidable* MSO*-theory, then $\langle \mathbb{N}, +1, \Sigma s(\mathbb{N})\rangle$ has a decidable* MSO*-theory.*

Theorem 7. *If $s \in \mathbb{S}_k^{\boldsymbol{N}}$, with $\boldsymbol{N} = (N_1, \ldots, N_m)$ such that $\langle \mathbb{N}, +1, N_1, \ldots, N_m\rangle$ has a decidable* MSO*-theory, then $\langle \mathbb{N}, +1, \Sigma s(\mathbb{N}), \Sigma s(N_1), \ldots, \Sigma s(N_m)\rangle$ has a decidable* MSO*-theory.*

3.2 Differentiably, k-computable sequences

The particular form of the predicates $\Sigma s(\mathbb{N})$ considered in Theorems 5, 6 and 7 leads naturally to the study of the following class of sequences.

Definition 9. *Let $k \geq 2$ and $\boldsymbol{N}$ a vector of subsets of $\mathbb{N}$. We define the class $\Sigma\mathbb{S}_k^{\boldsymbol{N}} \subseteq \mathbb{N}^{\mathbb{N}}$ as the set*

$$\Sigma\mathbb{S}_k^{\boldsymbol{N}} = \{\Sigma v \mid v \in \mathbb{S}_k^{\boldsymbol{N}}\}.$$

Theorem 5 means that for every sequence s in $\Sigma\mathbb{S}_k$, the structure $\langle \mathbb{N}, +1, s(\mathbb{N})\rangle$ has a decidable MSO-theory. In the same way, by Theorem 6 if $s \in \Sigma\mathbb{S}_k^{\boldsymbol{N}}$, and $\langle \mathbb{N}, +1, N_1, \ldots, N_m\rangle$ has a decidable MSO-theory, then $\langle \mathbb{N}, +1, s(\mathbb{N})\rangle$ has a decidable MSO-theory. Obviously, from Theorem 7, we obtain:

Corollary 1. *Let $v_1, \ldots, v_m \in \Sigma\mathbb{S}_k$, the following structure has a decidable MSO theory:*

$$\langle \mathbb{N}, +1, v_m(\mathbb{N}), v_m(v_{m-1}(\mathbb{N})), \ldots, v_m(v_{m-1}(\ldots(v_1(\mathbb{N}))))\rangle.$$

Proposition 4. *If P is a polynomial with positive integer coefficients, the sequence u defined by $u(n) = P(n)$ for all $n \geq 0$ belongs to $\Sigma\mathbb{S}_2$.*

Proposition 5. *Let s be a strictly increasing integer sequence, the sequence s^{-1} belongs to $\Sigma\mathbb{S}_2^{s(\mathbb{N}^+)}$.*

Corollary 2. *The two following structures have a decidable MSO-theory:*

$$\langle \mathbb{N}, +1, \{n^{k_m}\}_{n\geq 0}, \{n^{k_m k_{m-1}}\}_{n\geq 0}, \ldots, \{n^{k_m \cdots k_1}\}_{n\geq 0}\rangle, \text{ with } k_1, \ldots, k_m \geq 0,$$
$$\langle \mathbb{N}, +1, v_m(\mathbb{N}), v_{m-1}(\mathbb{N}), \ldots, v_1(\mathbb{N})\rangle, \text{ with } v_1(n) = 2^n \text{ and } v_{i+1}(n) = 2^{v_i(n)}.$$

We show now that classes $\Sigma\mathbb{S}_k^{\boldsymbol{N}}$ are closed by many operations.

Theorem 8.
0- For every $u \in \Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$, $k \geq 1$, and every integer $c \in \mathbb{N}$, the sequences Eu, $u + \frac{c}{1-X}$ (adding c to every term), belong to $\Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$;
if $u(n) \geq c$ then $u - \frac{c}{1-X}$ (subtracting c to every term) belongs to $\Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$;
if $u(0) \geq c$, then the sequence $0 \mapsto c, n+1 \mapsto u(n)$ belongs to $\Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$.
1- For every $u, v \in \Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$, $k \geq 1$, the sequence $u + v$ belongs to $\Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$.
2- For every $u, v \in \Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$, $k \geq 2$, the sequence $u \odot v$ belongs to $\Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$.
3- For every $u \in \Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$, $v \in \Sigma\mathbb{S}_k$, $k \geq 2$, $u \times v$ belongs to $\Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$.
4- For every $u \in \Sigma\mathbb{S}_k$, $k \geq 2$,such that $v(0) \geq 1$, the sequence u defined by: $u(0) = 1$ and $u(n+1) = \sum_{m=0}^{n} u(m) \cdot v(n-m)$ (the convolution inverse of $1 - Xv$) belongs to $\Sigma\mathbb{S}_{k+1}$.
5- For every $u \in \Sigma\mathbb{S}_k$, $v \in \Sigma\mathbb{S}_\ell^{\boldsymbol{N}}$, $k, l \geq 2$, $u \circ v$ belongs to $\Sigma\mathbb{S}_{k+\ell-1}^{\boldsymbol{N}}$.
6- For every $k \geq 2$, if $u_1(n), \ldots u_p(n)$ is the vector of solutions of a system of recurrent equations expressed by polynomials in $\Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}[X_1, \ldots, X_p]$, with initial conditions $u_i(0), u_i(1) \in \mathbb{N}$, with $u_i(0) \leq u_i(1)$, then $u_1 \in \Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$.

Corollary 3. *Let t be the sequence defined by $t(n) = P(n)s^{-1}(n)^\ell$, where $s \in \mathbb{S}_k$ is strictly growing sequence, P is a polynomial with positive integer coefficients and ℓ is a positive integer. Then the structure $\langle \mathbb{N}, +1, t(\mathbb{N})\rangle$ has a decidable* MSO*-theory.*

Corollary 4. *Structures $\langle \mathbb{N}, +1, (n\lfloor\sqrt{n}\rfloor)_{n\in\mathbb{N}}\rangle$, and $\langle \mathbb{N}, +1, (n\lfloor \log n\rfloor)_{n\geq 1}\rangle$ have a decidable* MSO*-theory.*

Remark 1. It can be proved that classes $\Sigma\mathbb{S}_k$ are included in the class of "residually ultimately periodic" (RUP) sequences studied by [3]. It is shown in [3] that for any RUP sequence s , the theory of $\langle \mathbb{N}, +1, s(\mathbb{N})\rangle$ is decidable. It can be proved that sequences in $\Sigma\mathbb{S}_k^{\boldsymbol{N}}$ considered Theorem 6, like $(n\lfloor\sqrt{(n)}\rfloor)_{n\in\mathbb{N}}$ or $(n\lfloor \log(n)\rfloor)_{n\in\mathbb{N}}$ are not RUP.

Acknowledgements This work is a part of the Ph.D. of the author. The author thanks her supervisors, G. Sénizergues and F. Carrère for having directed and allowed this work.

References

1. A. V. Aho. Nested stack automata. *J. Assoc. Comput. Mach.*, 16:383–406, 1969.
2. A. Bès. A survey of arithmetical definability. *Bull. Belg. Math. Soc. Simon Stevin*, (suppl.):1–54, 2001. A tribute to Maurice Boffa.
3. O. Carton and W. Thomas. The monadic theory of morphic infinite words and generalizations. *Inform. and Comput.*, 176(1):51–65, 2002.
4. D. Caucal. On infinite terms having a decidable monadic theory. In *Mathematical foundations of computer science 2002*, volume 2420 of *Lecture Notes in Comput. Sci.*, pages 165–176. Springer, Berlin, 2002.

5. W. Damm. The IO- and OI-hierarchies. *Theoret. Comput. Sci.*, 20(2):95–207, 1982.
6. W. Damm and A. Goerdt. An automata-theoretical characterization of the OI-hierarchy. *Inform. and Control*, 71(1-2):1–32, 1986.
7. Calvin C. E. and M. O. Rabin. Decidability and undecidability of extensions of second (first) order theory of (generalized) successor. *J. Symbolic Logic*, 31(2):169–181, 1966.
8. J. Engelfriet. Iterated pushdown automata and complexity classes. In *Proceedings of the 14th Symposium on Theory of Computing*, pages 365–373. Association for Computing Machinery, 1983.
9. J. Engelfriet and G. Slutzki. Extended macro grammars and stack controlled machines. *J. Comput. System Sci.*, 29(3):366–408, 1984.
10. J. Engelfriet and H. Vogler. Corrigenda: "Pushdown machines for the macro tree transducer". *Theoret. Comput. Sci.*, 48(2-3):339 (1987), 1986.
11. S Fratani. Regular sets over tree structures. Submitted to TCS, Available at url: http://www.cmi.univ-mrs.fr/˜sfratani/.
12. S Fratani. *Automates à piles de piles ... de piles*. PhD thesis, Université Bordeaux 1, 2005.
13. S. Fratani and G. Sénizergues. Iterated pushdown automata and sequences of rational numbers. In *Ann. Pure Appl. Logic*, volume 141, pages 363–411. Elsevier, 2005.
14. S. A. Greibach. Full AFLs and nested iterated substitution. *Inform. and Control*, 16:7–35, 1970.
15. M.A. Harrison. *Introduction to Formal Language Theory*. Addison-Wesley, Reading, Mass., 1978.
16. F.A Hosch. *Decision Problems in Büchi's Sequential Calculus*. PhD thesis, University of New Orleans, Luisiana, 1971.
17. T. Knapik, D. Niwiński, and P. Urzyczyn. Higher-order pushdown trees are easy. In *Foundations of software science and computation structures (Grenoble, 2002)*, volume 2303 of *Lecture Notes in Comput. Sci.*, pages 205–222. Springer, Berlin, 2002.
18. A. Maes. An automata-theoretic decidability proof for first-order theory of $\langle \mathbf{N}, <, P \rangle$ with morphic predicate *P*. *J. Autom. Lang. Comb.*, 4(3):229–245, 1999. Journées Montoises d'Informatique Théorique (Mons, 1998).
19. A. N. Maslov. Multilevel pushdown automata. *Problemy Peredači Informacii*, 12(1):55–62, 1976.
20. M.O. Rabin. Decidable theories. *Handbook of Mathematical Logic*, pages 595–629, 1977.
21. A. L. Semenov. Decidability of monadic theories. In *Mathematical foundations of computer science, 1984 (Prague, 1984)*, volume 176 of *Lecture Notes in Comput. Sci.*, pages 162–175. Springer, Berlin, 1984.
22. D. Siefkes. Decidable extensions of monadic second order successor arithmetic. In *Automatentheorie und formale Sprachen (Tagung, Math. Forschungsinst., Oberwolfach, 1969)*, pages 441–472. Bibliographisches Inst., Mannheim, 1970.

4 Annexe

4.1 Some basic tools

Let $\mathcal{A} = (Q, \Sigma, \mathcal{A}_k, \boldsymbol{N}, \Delta, q_0, Z)$ be some k-cpda. A total state of $\mathcal{A}$ is any pair $(q, \omega) \in Q \times k\text{-}\mathsf{pds}(\mathcal{A}_k)$.

If α is used to denote a symbol of Σ, then α_ε denotes the letter α or the empty word.

Derivation We associate with $\mathcal{A}$ an infinite "alphabet"

$$V_{\mathcal{A}} = \{(p, \omega, q) \mid p, q \in Q,\ \omega \in k\text{-}\mathsf{pds}(\mathcal{A}_k) - \{\varepsilon\}\}, \tag{1}$$

and a set of *productions* associated with $\mathcal{A}$, denoted $P_{\mathcal{A}}$ and made of the set of all the following rules:

– the *transition* rules:

$$(p, \omega, q) \vdash_{\mathcal{A}} \alpha_\varepsilon(p', \omega', q) \text{ if } (p, \alpha_\varepsilon, \omega) \to_{\mathcal{A}} (p', \varepsilon, \omega') \text{ and } q \in Q \text{ is arbitrary,}$$

$$(p, \omega, q) \vdash_{\mathcal{A}} \alpha_\varepsilon \text{ if } (p, \alpha_\varepsilon, \omega) \to_{\mathcal{A}} (q, \varepsilon, \varepsilon).$$

– the *decomposition* rules:

$$(p, \omega, q) \vdash_{\mathcal{A}} (p, \eta, r)(r, \eta', q)$$

if $\omega = \eta \cdot \eta', \eta \neq \varepsilon, \eta' \neq \varepsilon$ and $r \in Q$ is arbitrary.

The one-step *derivation* generated by $\mathcal{A}$, denoted by $\vdash_{\mathcal{A}}$, is the smallest subset of $(V \cup \Sigma)^* \times (V \cup \Sigma)^*$ which contains $P_{\mathcal{A}}$ and is compatible with left product and right product. Finally, the *derivation* generated by $\mathcal{A}$, denoted $\vdash^*_{\mathcal{A}}$, is the reflexive and transitive closure of $\vdash_{\mathcal{A}}$. These notions correspond to the usual notion of *context-free grammar* associated with the following automaton of level $\mathcal{A}_1$: this automaton has the pushdown alphabet $A = \{a[\omega] \mid a \in A_k,\ \omega \in (k-1)\text{-}\mathsf{pds}\}$ and has the transition function

$$\Delta_1(q, \alpha_\varepsilon, a[\omega]) = \{(\eta', q') \in Q \times A^* \mid (q, \alpha, a[\omega]) \to_{\mathcal{A}} (q', \varepsilon, \eta')\}.$$

Of course, as soon as $k \geq 2$, this pushdown alphabet is infinite, but all the usual properties of the relation $\vdash_{\mathcal{A}} = \vdash_{\mathcal{A}_1}$ and its links with $\to_{\mathcal{A}} = \to_{\mathcal{A}_1}$ remains true in this context (see [15, proof of the Theorem 5.4.3, pp 151-158]). In particular, for every $\sigma \in \Sigma^*, p, q \in Q, \omega \in A^*$,

$$(p, \omega, q) \vdash^*_{\mathcal{A}} \sigma \Leftrightarrow (p, \sigma, \omega) \xrightarrow{*}_{\mathcal{A}} (q, \varepsilon, \varepsilon).$$

The following lemma is useful.

Lemma 1. *Let $p_i, q_i \in Q$, $\omega_i \in A^*$ for $i \in \{1, 2, 3\}$. The following properties are equivalent:*

1. $(p_1, \omega_1, q_1) \vdash^*_{\mathcal{A}} (p_2, \omega_2, q_2)(p_3, \omega_3, q_3)$
2. there exists $\omega_2', \omega_3' \in A^*$*, such that:*

$$(p_1, \varepsilon, \omega_1) \xrightarrow{*}_{\mathcal{A}} (p_2, \varepsilon, \omega_2\omega_2'),$$
$$(q_2, \varepsilon, \omega_2') \xrightarrow{*}_{\mathcal{A}} (p_3, \varepsilon, \omega_3\omega_3') \text{ and}$$
$$(q_3, \varepsilon, \omega_3') \xrightarrow{*}_{\mathcal{A}} (q_1, \varepsilon, \varepsilon).$$

We usually assume that pushdown alphabets and Q are disjoint, therefore, omitting the commas in (p, ω, q) does not lead to any confusion.

Terms Let us fix a family $(\mathcal{I}_k)_{k \geq 0}$ of denumerable sets of symbols: $\mathcal{I}_k = \{\Omega, \Omega', \Omega'', \ldots, \Omega_1, \Omega_2, \ldots\}$ denotes the set of **indeterminates** of level k. We suppose that $\mathcal{I}_k \cap \mathcal{I}_i = \emptyset$ for all $i, j \geq 0$ and that pushdown alphabets and sets of indeterminates are always disjoint. A k-term is a k-pds in which are added symbols that do not belong to the pushdown alphabets. Each indeterminate of level i (i.e., in $\mathcal{I}_i$) can be place anywhere at the level i of a term. Let us define inductively the set $\mathcal{T}_k(\mathcal{A}_k)$ of terms of level k, for $k \geq 0$:

– $\mathcal{T}_0(\mathcal{A}_0) = \{\varepsilon\}$
– $\mathcal{T}_{k+1}(\mathcal{A}_{k+1}) = (A_{k+1}[\mathcal{T}_k(\mathcal{A}_k)] \cup \mathcal{I}_{k+1})^*$.

We denote a k-term T by $T[\Omega_1, \ldots, \Omega_n]$ provided that the only indeterminates appearing in T are $\Omega_1, \ldots, \Omega_n$.

The concatenation product over k-pds is generalized to $\mathcal{T}_k$, so as the operation top and the instructions push, pop et change.

For all term T such that $\text{top}_i(T)$ is an indeterminate, the level i instructions push_{a_i}, pop_i and change_{a_i} are undefined, else, they are defined as for k-pds.

Substitutions Given $T[\Omega_1, \ldots, \Omega_n] \in \mathcal{T}_k(\mathcal{A}_k)$ with $\Omega_i \in \mathcal{I}_{k_i}$ for $i \in [1, n]$, $k_i \in [1, k]$ and $T_1 \in \mathcal{T}_{k_1}, \ldots, T_n \in \mathcal{T}_{k_n}$, we denote by $T[T_1, \ldots, T_n]$ the k-term obtained by substituting T_i for Ω_i.

The following "substitution principle" is straightforward and will be widely used in our proofs. Given $\mathcal{A} \in k\text{-}\mathsf{CPDA}^{\boldsymbol{N}}$, we extend the relations $\vdash_{\mathcal{A}}$ and $\to \mathcal{A}$ to terms that do not contain indeterminates of level 1.

Lemma 2. *Given* $\mathcal{A} \in k\text{-}\mathsf{CPDA}^{\boldsymbol{N}}$ *and* $\boldsymbol{\Omega} = (\Omega_1, \ldots, \Omega_n)$ *where each* Ω_i *is an indeterminate of level* $k_i \in [2, k]$*. If* $T[\boldsymbol{\Omega}]$ *and* $T'[\boldsymbol{\Omega}]$ *are two terms in* $\mathcal{T}_k(A_1, \ldots, A_k)$*, then for all* p*,* q*,* p'*,* $q' \in Q$*,*

$$\text{if } (pT[\boldsymbol{\Omega}]q) \to^*_{\mathcal{A}} (p'T'[\boldsymbol{\Omega}]q'), \text{ then}$$

– *for all* $\boldsymbol{H} = (H_1, \ldots, H_n)$ *such that for all* $i \in [1, n]$*,* H_i *is a* k_i*-term,*

$$(pT[\boldsymbol{H}]q) \to^*_{\mathcal{A}} (p'T'[\boldsymbol{H}]q'),$$

– *for all* $\boldsymbol{\omega} = (\omega_1, \ldots, \omega_n)$ *such that for all* $i \in [1, n]$, ω_i *is a* k_i*-pds,*

$$(pT[\boldsymbol{\omega}]q) \to_{\mathcal{A}}^{*} (p'T'[\boldsymbol{\omega}]q').$$

The key idea for this lemma is that, as $A_i \cap \mathcal{I}_i = \emptyset \ \forall i \geq 1$, the symbols Ω_i can be copied or erased during the derivation but they cannot *influence* the sequence of rules uses in that derivation.

4.2 Proof of Proposition 3

Proof. Let $\mathcal{A} = (\{q_0\}, \{\alpha\}, (\{a_1\}, \{a_2\}), s(\mathbb{N}^+), \Delta, q_0, a_2)$ with $\Delta(q_0, \varepsilon, a_2, 0) = (q_0, \alpha, a_2, 1) = (\text{pop}_2, q_0)$ and $\Delta(q_0, \varepsilon, a_2a_1, 0) = \Delta(q_0, \alpha, a_2a_1, 1) = (\text{pop}_1, q_0)$.

Starting from a configuration $(q_0, \sigma, a_2[{a_1}^n])$, $\mathcal{A}$ pops iteratively the counter, by reading to each iteration a terminal letter α iff the counter belongs to $s(\mathbb{N}+)$. Finally, when the stack remains empty, the length of the read terminal word is the number of elements of $[0, n] \cap s(\mathbb{N}^+)$, i.e., $s^{-1}(n)$.

4.3 Proof of Theorem 4

In order to simplify the proofs, we will often use in automata, some transition of the following form: $(q, \sigma, w, \text{instr}_1 \ldots \text{instr}_m, p)$ where $\sigma \in \Sigma^*$, $m \geq 1$ and each $instr_i$ is a pushdown instruction. A such a transition is applied in the following way:

$(q, \sigma\sigma', \omega) \to (p, \sigma'\omega')$ iff $\text{top}(\omega) = w$ and $\omega' = \text{instr}_m(\cdots(\text{instr}_1(\omega))$.

The same extension will be used for controlled automata. Clearly, we do not modify the expressiveness of a class of automata by using this kind of transitions. In the same way, if there exists a deterministic automaton in k-$\mathsf{DCPDA}^{\boldsymbol{N}}$ using such transitions, then one can construct a deterministic "standard" automaton in k-$\mathsf{DCPDA}^{\boldsymbol{N}}$ recognizing the same language.

In all this section, we will use the following notation:
for all $k \geq 2$, $i \in [2, k+1]$,

$$T_{k,i}[\Omega_{i-1}] := a_k[a_{k-1}[\cdots[a_i[\Omega_{i-1}]]\cdots]],$$

for the precise symbols $a_1, \ldots, a_k$. In particular, $T_{k,k}[\Omega_{k-1}] = a_k[\Omega_{k-1}]$ and $T_{k,k+1}[\Omega_k] = \Omega_k$.

We start by giving two lemmas which will be widely use in the next constructions. They are in fact two versions of the same lemma, a weak version and a strong version, which allows, from a an automaton in k-DCPDA$^{\boldsymbol{N}}$ computing a sequence s, to construct a new automaton in k-DCPDA$^{\boldsymbol{N}}$ making $s(n)$ copies of a particular configuration. We construct this automaton in a such way as it is ready to be composed with another.

Lemma 3 (Weak normal form). *Let* s *be a sequence of natural numbers,* $k \geq 1$ *and* $\mathcal{A} \in (k+1)$-$\mathsf{DCPDA}^{\boldsymbol{N}}$ *defined over the pushdown alphabets* $A_1, \ldots, A_{k+1}$ *where* $a_1 \in A_1$,$\ldots$, $a_{k+1} \in A_{k+1}$ *and fulfilling,*

(**H1**) $\forall n \geq 0,\ (q_0, \alpha^{s(n)}, a_{k+1}[a_k[\ldots[a_2[a_1{}^n]]\ldots]]) \xrightarrow{*}_{\mathcal{A}} (q_0, \varepsilon, \varepsilon)$.
(**H2**) *$\mathcal{A}$ does not contain lefthand side of the form (q, α, ε) or $(q, \varepsilon, \varepsilon)$.*

Then, we can construct $\mathcal{B} \in (k+1)$-$\mathsf{DCPDA}^{\boldsymbol{N}}$ defined on the pushdown alphabets $A_1 \cup A'_1, \ldots, A_{k+1} \cup A_{k+1}{}'$, where A'_{k+1} contains a special symbol A_{k+1}, whose set of states contains q_0 and such that:
(**P1**) $(q_0, a_{k+1}[a_k[\ldots[a_2[a_1{}^n]]\ldots]], q_0) \vdash^*_{\mathcal{B}} (q_0, \mathrm{A}_{k+1}[\varepsilon], q_0)^{s(n)}$.
(**P2**) *Δ' does not contain lefthand side of the form $(q_0, \varepsilon, \varepsilon)$.*
(**P3**) *Δ' does not contain lefthand side of the form $(q_0, \varepsilon, \mathrm{A}_{k+1} \cdot w)$.*

Construction: Suppose that $\mathcal{A} = (Q, \{\alpha\}, (A_1, \ldots, A_{k+1}), \boldsymbol{N}, \Delta, q_0, a_{k+1})$ is an automaton fulfilling hypothesis (**H1**), (**H2**). Let $B_{k+1} = A_{k+1} \cup \{\mathrm{A}_{k+1}, \mathrm{B}_{k+1}\} \cup \{(b_{k+1}, \delta) \mid b_{k+1} \in A_{k+1},\ \delta \in \Delta\}$ and

$$\mathcal{B} = (Q, \emptyset, (A_1, \ldots, A_k, B_{k+1}), \boldsymbol{N}, \Delta', q_0, a_{k+1})$$

where Δ' consists of the following transitions:

- for all $\Delta(p, \varepsilon, w, \boldsymbol{o}) = (\mathrm{instr}, q)$,
  (1) $\Delta'(p, \varepsilon, w, \boldsymbol{o}) = (\mathrm{instr}, q)$,
- for all $b_{k+1} \in A_{k+1}$ and $\delta = (p, \alpha, b_k w, \boldsymbol{o}, \mathrm{instr}, q) \in \Delta$,
  (2.1) $\Delta'(p, \varepsilon, b_{k+1} w, \boldsymbol{o}) = (\mathrm{change}_{(b_{k+1}, \delta)}\ \mathrm{push}_{\mathrm{B}_{k+1}}, q_0)$,
  (2.2) $\Delta'(q_0, \varepsilon, (b_{k+1}, \delta) w, \boldsymbol{o}) = (\mathrm{change}_{b_{k+1}}\ \mathrm{instr}, q)$,
- for all $w \neq \varepsilon \in \mathrm{top}(k\text{-}\mathsf{pds}(\mathcal{A}_k))$, $\boldsymbol{o} \in \{0,1\}^{|\boldsymbol{N}|}$,
  (3.1) $\Delta'(q_0, \varepsilon, \mathrm{B}_{k+1} w, \boldsymbol{o}) = (\mathrm{pop}_k, q_0)$,
- for all $\boldsymbol{o} \in \{0,1\}^{|\boldsymbol{N}|}$,
  (3.2) $\Delta'(q_0, \varepsilon, \mathrm{B}_{k+1}, \boldsymbol{o}) = (\mathrm{push}_{\mathrm{A}_{k+1}}, q_0)$.

Proof. Let us prove the validity of the construction.
Determinism and conditions (**P2,P3**): Let us verify that $\mathcal{B}$ is deterministic. The automaton $\mathcal{A}$ being deterministic, two distinct transitions of types 1 or 2 are always incompatible. Transitions of type 3 are incompatible and since B_{k+1} is a new symbol, each of them is incompatible with all transition of type 1 or 2. Then $\mathcal{B}$ is deterministic.

The automaton $\mathcal{A}$ fulfilling hypothesis (**H2**), it is obvious that $\mathcal{B}$ fulfills (**P2**). Finally, the condition (**P3**) is verified by transitions resulting from $\mathcal{A}$ (type 1 and 2) since A_{k+1} is a new symbol, the since we do not have added transitions using this symbol, the condition (**P3**) is verified by $\mathcal{B}$.
Condition (**P1**): In order to prove that $\mathcal{B}$ fulfills the condition (**P1**), we establish the two following implications:

for all $p, q \in Q$, $\omega, \omega' \in k+1\text{-}\mathsf{pds}(A_1, \ldots, A_k, B_{k+1})$

$$(p, \varepsilon, \omega) \xrightarrow{*}_{\mathcal{A}} (q, \varepsilon, \omega') \implies (p\omega q_0) \vdash^*_{\mathcal{B}} (q\omega' q_0), \tag{2}$$
$$(p, \alpha, \omega) \to_{\mathcal{A}} (q, \varepsilon, \omega') \implies (p\omega q_0) \vdash^*_{\mathcal{B}} (q_0 \mathrm{A}_{k+1}[\varepsilon] q_0)(q\omega' q_0). \tag{3}$$

Note that we let open the possibility that ω, ω' contain occurences of letters that do not belong to A_{k+1}. The relation $\xrightarrow{*}_{\mathcal{A}}$ is defined from transitions of $\mathcal{A}$, but applied to total states in $Q \times k\text{-}\mathsf{pds}(A_1, \ldots, A_k, B_{k+1})$.

The implication (2) is obtained by translation, in terme of derivation, of transitions of type (1). Let us prove (3). We suppose that $\omega = b_{k+1}[\omega_1]\omega''$, $\omega' = \mathrm{instr}(\omega)$ and $(p, \alpha, \omega) \to_\delta (q, \varepsilon, \omega')$, for $\delta \in \Delta$. The following derivation holds:

$$\begin{aligned}(p\omega q_0) &\vdash_{\mathcal{B}} (q_0 \mathrm{B}_{k+1}[\omega_1](b_{k+1}, \delta)[\omega_1]\omega'' q_0) \text{ (by transitions (2.1))}\\ &\vdash^*_{\mathcal{B}} (q_0 \mathrm{B}_{k+1}[\varepsilon](a, \delta)[\omega_1]\omega'' q_0) \text{ (by iteration of transitions (3.1))}\\ &\vdash_{\mathcal{B}} (q_0 \mathrm{A}_{k+1}[\varepsilon](a, \delta)[\omega_1]\omega'' q_0) \text{ (by transitions (3.2))}\\ &\vdash_{\mathcal{B}} (q_0 \mathrm{A}_{k+1}[\varepsilon]q_0)(q_0(a, \delta)[\omega_1]\omega'' q_0) \text{ (by decomposition rule)}\\ &\vdash_{\mathcal{B}} (q_0 \mathrm{A}_{k+1}[\varepsilon]q_0)(q\omega' q_0)\text{(by transitions (2.2))}.\end{aligned}$$

By using implications (2) and (3), and hypothesis (**H1**), an obvious induction on the length of the derivation (**H1**) proves that for all $n \geq 0$,

$$(q_0 a_{k+1}[\cdots[a_1{}^n]\cdots]q_0) \vdash^*_{\mathcal{B}} (q_0 \mathrm{A}_{k+1}[\varepsilon]q_0)^{s(n)}.$$

Lemma 4 (Strong normal form). *Let s be a sequence of natural numbers, $k \geq 2$ and $\mathcal{A} \in (k+1)\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}$ defined over alphabets $A_1, \ldots, A_{k+1}$ where $a_1 \in A_1, \ldots, a_{k+1} \in A_{k+1}$ and fulfilling $\forall n \geq 0$,*
(**H1**) $(q_0, \alpha^{s(n)}, a_{k+1}[a_k[\ldots[a_2[a_1{}^n]]\ldots]]) \xrightarrow{*}_{\mathcal{A}} (q_0, \varepsilon, \varepsilon)$.
(**H2**) *$\mathcal{A}$ does not contain lefthand side of the form $(q_0, \alpha, \varepsilon)$ or $(q_0, \varepsilon, \varepsilon)$.*

Then, we can construct $\mathcal{B} \in (k+1)\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}$ defined over the alphabets $A_1 \cup A_1', \ldots, A_{k+1} \cup A_{k+1}{}'$, where A'_{k+1} contains a special symbol A_{k+1}, whose set of states contains q_0 and such that:
(**Q1**) $\forall \Omega_k \in \mathcal{I}_k,\ (q_0, a_{k+1}[a_k[\ldots[a_2[a_1{}^n]]\ldots]\Omega_k], q_0) \vdash^*_{\mathcal{B}} (q_0, \mathrm{A}_{k+1}[\Omega_k], q_0)^{s(n)}$
(**Q2**) *Δ' does not contain lefthand side of the form $(q_0, \varepsilon, \varepsilon)$.*
(**Q3**) *Δ' does not contain lefthand side of the form $(q_0, \varepsilon, \mathrm{A}_{k+1} \cdot w)$.*

Proof. Let us consider the following derivations:
Initialization rule (D0):

$$(q_0 a_{k+1}[T_{k,2}[a_1{}^n]\Omega_k]q_0) \vdash^*_{\mathcal{B}} (q_0 b_{k+1}[T_{k,2}[a_1{}^n]\mathrm{B}_k[T_{k-1,2}[a_1{}^n]]\Omega_k]q_0)$$

s-computation (D1):

$$(q_0 b_{k+1}[T_{k,2}[a_1{}^n]\mathrm{B}_k[T_{k-1,2}[a_1{}^n]]\Omega_k]q_0) \vdash^*_{\mathcal{B}} (q_0 \mathrm{A}_{k+1}[\mathrm{B}_k[T_{k-1,2}[a_1{}^n]]\Omega_k]q_0)^{s(n)}$$

Ending rule (D2):

$$(q_0 \mathrm{A}_{k+1}[b_k[T_{k-1,2}[a_1{}^n]]\Omega_k]q_0) \vdash^*_{\mathcal{A}'} (q_0 \mathrm{A}_{k+1}[\Omega_k]q_0)$$

If $\mathcal{B}$ is an automaton for which these derivations hold, then the following derivation (**Q1**) is valid:

$$\begin{aligned}(q_0, a_{k+1}[T_{k,2}[a_1{}^n]\Omega_k], q_0) &\vdash^*_{\mathcal{B}} (q_0 b_{k+1}[T_{k,2}[a_1{}^n]\mathrm{B}_k[T_{k-1,2}[a_1{}^n]]\Omega_k]q_0)\\ &\vdash^*_{\mathcal{B}} (q_0 \mathrm{A}_{k+1}[b_k[T_{k-1,2}[a_1{}^n]]\Omega_k]q_0)^{s(n)}\\ &\vdash^*_{\mathcal{B}} (q_0 \mathrm{A}_{k+1}[\Omega_k]q_0)^{s(n)}.\end{aligned}$$

To prove the lemma, we just have to construct a deterministic automaton $\mathcal{B}$ for which derivations (D0), (D1) et (D2) hold and fulfilling conditions (**Q2,Q3**). *Construction*: By using Lemma 3, and a suitable renaming of the pushdown alphabets, we obtain an deterministic automaton $\mathcal{A} = (Q, \emptyset, \mathcal{A}_{k+1}, \boldsymbol{N}, \Delta', q_0)$ fulfilling conditions $(\mathbf{P1}(b_{k+1}a_k \cdots a_1, \mathrm{A}_{k+1})), (\mathbf{P2})$ and $(\mathbf{P3}(\mathrm{A}_{k+1}))$.

Consider $\mathcal{B} = (Q, \emptyset, (B_1, \ldots, B_{k+1}), \boldsymbol{N}, \Delta \cup \Delta', q_0)$, with $B_{k+1} = A_{k+1} \cup \{a_{k+1}\}$, $B_k = A_k \cup \{\mathrm{B}_k\}$, $B_i = A_i$ for $1 \leq i \leq k-1$, and Δ consists of the following transitions:

- for symbols $a_1, a_2, \ldots a_k, b_{k+1}$ used in (**P1**), for all $\boldsymbol{o} \in \{0,1\}^{|\boldsymbol{N}|}$,
  (0) $\Delta(q_0, \varepsilon, a_{k+1} \cdots a_2 a_1, \boldsymbol{o}) = (\mathrm{change}_{b_{k+1}}\ \mathrm{change}_{\mathrm{B}_k}\ \mathrm{push}_{a_k}, q_0)$,
- for all $(q, \varepsilon, c_{k+1}, \chi_{\boldsymbol{N}}(0), \mathrm{instr}, p) \in \Delta'$, $c_{k+1} \in A_{k+1}$ unspecified, $\boldsymbol{o} \in \{0,1\}^{|\boldsymbol{N}|}$,
  (1) $\Delta(q, \varepsilon, c_{k+1}\mathrm{B}_k a_{k-1} \cdots a_2 a_1, \boldsymbol{o}) = \Delta(q, \varepsilon, c_{k+1}\mathrm{B}_k a_{k-1} \cdots a_2, \boldsymbol{o}) = (\mathrm{instr}, p)$,
- for all $w \in \mathrm{top}((k-1)\text{-}\mathsf{pds}(A_1, \ldots, A_{k-1}))$, $\boldsymbol{o} \in \{0,1\}^{|\boldsymbol{N}|}$,
  (2) $\Delta'(q_0, \varepsilon, \mathrm{A}_{k+1}\mathrm{B}_k w, \boldsymbol{o}) = (\mathrm{pop}_k, q_0)$.

Determinism and conditions (**Q2, Q3**): Automaton $\mathcal{A}$ is deterministic and since a_{k+1}, B_k are new symbols, the addition of transitions (0) and (2) does not introduce non-determinism. In the same way, transitions of type (1) are incompatibles with all transitions of Δ' or with transitions of another type, and since $\mathcal{A}$ is deterministic, for all pair $(q, c_{k+1}) \in Q \times A_{k+1}$, there exists a unique transition whose lefthand side is $(q, \varepsilon, c_{k+1}, \chi_{\boldsymbol{N}}(0))$ and transitions of type (1) are then all incompatibles between them. $\mathcal{B}$ is then deterministic. In addition, $\mathcal{A}$ verifies (**P2**) and $(\mathbf{P3}(\mathrm{A}_{k+1}))$, and the addition of transitions (0),(1), and (2), preserve these properties. Then $\mathcal{B}$ verifies (**Q2**) and $(\mathbf{Q3}(\mathrm{A}_{k+1}))$.
Condition (**Q1**): From the discussion preceding the construction, we just have to show that derivations (D0), (D1) and (D2) are realized by $\mathcal{B}$. The derivation (D0) is obtained by application of a transition of type (0), and (D2) is realized by a transition of type (2). It rest then to verify that (D1) is a valid derivation.

Let us define for all $n \geq 0$, the application

$$\tau_n : (k+1)\text{-}\mathsf{pds}(A_1, \ldots, A_{k+1}) \to \mathcal{T}_k(B_1, \ldots, B_{k+1})$$

associating to any $(k+1)$-pds, the term obtained by adding $\mathrm{B}_k[T_{k-1,2}[{a_1}^n]]\Omega_k$ at the bottom of each of them k-pds:

- $\forall \omega = c_{k+1}[\omega_1]\omega' \in (k+1)\text{-}\mathsf{pds}$, $\tau_n(\omega) = c_{k+1}[\omega_1 \mathrm{B}_k[T_{k-1,2}[{a_1}^n]]\Omega_k]\tau_n(\omega)$,
- $\tau_n(\varepsilon) = \varepsilon$.

For all $\omega, \omega' \in (k+1)\text{-}\mathsf{pds}$, $p, q \in Q$, $n \geq 0$

$$(p, \varepsilon, \omega) \to_{\mathcal{A}} (q, \varepsilon, \omega') \implies (p, \varepsilon, \tau_n(\omega)) \to_{\mathcal{B}} (q, \varepsilon, \tau_n(\omega')). \qquad (4)$$

The property can be easily verified:

- if $\mathrm{top}_k(\omega) \neq \varepsilon$, then $\mathrm{top}_k(\omega) = \mathrm{top}_k(\tau_n(\omega))$ and the transition applied to the lefthand side of the implication (4) is also applicable to $(p, \varepsilon, \tau_n(\omega))$ then

$$(p, \varepsilon, \tau_n(\omega)) \to_{\mathcal{B}} (q, \varepsilon, \tau_n(\omega')),$$

– else, $\omega = c_{k+1}[\varepsilon]\omega'$, then the instruction applied to the lefthand side of (4) has inevitably the form $(p, \varepsilon, c_{k+1}, \chi_{\boldsymbol{N}}(0), \mathrm{instr}, q)$ where instr is whether a $(k+1)$-instruction, or an instruction push of level k. Then, there exists $\boldsymbol{o} = \chi_{\boldsymbol{N}}(n)$, such that the transition of type (2) $(p, \varepsilon, b_{k+1}\mathrm{B}_k \cdots a_1, \boldsymbol{o}, \mathrm{instr}, q)$ belongs to Δ and

$$(p, \varepsilon, \tau_n(\omega)) \rightarrow_{\mathcal{B}} (q, \varepsilon, \tau_n(\omega')).$$

Let us reformulate these results in term of derivations:
for all $\omega, \omega_i \in (k+1)$-$\mathsf{pds}$, $i \in [1, \ell]$, $p, q, q_i, p_i \in Q$, $n, m \geq 0$:

$$(p, \omega, q) \vdash^m_{\mathcal{A}} \prod_{i=1}^{\ell} (p_i, \omega_i, q_i) \implies (p, \tau_n(\omega), q) \vdash^m_{\mathcal{B}} \prod_{i=1}^{\ell} (p_i, \tau_n(\omega_i), q_i) \qquad (5)$$

This implication can be easily verified by an induction over $m \geq 0$. If the applied rule is a decomposition rule, it also applies to $(p, \tau_n(\omega), q)$ and the property is then verified. If the rule applied comes from a transition, then (4) implies (5).

We can now achieve the proof of the lemma by showing that $\mathcal{B}$ realize the derivation (D1). By substituting the derivation $(\mathbf{P1}(b_{k+1}a_k \cdots a_1, \mathrm{A}_{k+1}))$ to the lefthand side of (5), we obtain

$$(q_0, \tau_n(b_{k+1}[T_{k,2}[{a_1}^n]]), q_0) \vdash^*_{\mathcal{A}'} (q_0, \tau_n(\mathrm{A}_{k+1}[\varepsilon]) q_0)$$

i.e., (D1).

Remark 2.

1. Let us add to the transitions of $\mathcal{B}$ constructed in Lemma 3 (resp. 4), transitions $(q_0, \alpha, \mathrm{A}_{k+1}, \boldsymbol{o}, \mathrm{pop}_1, q_0)$ (for $\boldsymbol{o}$ unspecified), we obtain then a new automaton $\mathcal{B}'$ verifying hypothesis (**H0**), (**H1**) of Definition 7.
2. Properties (**P1**) (resp. (**Q1**)) make $\mathcal{B}$ ready to be combined with another automaton: it suffices to add transitions starting from $q_0\mathrm{A}_k[\omega]$ for ω well chosen, and leading to a configuration of another deterministic automaton. Properties (**P2**, **P3**) (resp. (**Q2**, **Q3**)) allow that the new automaton thus composed is deterministic.
3. The strong version of the lemma is valid only for $k \geq 2$. The case $k = 1$ is particular since the indeterminate has then level 1 and we have remarked that because of controllers, relations $\vdash_{\mathcal{A}}$ and $\rightarrow_{\mathcal{A}}$ are defines only for terms that do not contains indeterminates of level 1. In addition, if $k = 1$, condition (**Q1**) is written $(q_0, a_2[{a_1}^n \Omega_1], q_0) \vdash^*_{\mathcal{B}} (q_0, \mathrm{A}_2[\Omega_1], q_0)^{s(n)}$ and we see that it is not possible any more to insert the separator B_1 between a_1 and Ω_1, necessary to the proof of the strong version of the lemma.
4. The construction given for the weak version is completely independent of the chosen controllers, and it is not necessary to know their value to carry out the construction. For the strong version, the *effective* construction of the automaton requires to be able to compute the value of $\chi_{\boldsymbol{N}}(0)$.

Proposition 6 (Somme). *If $s, t \in \mathbb{S}^{\boldsymbol{N}}_{k+1}$ with $k \geq 1$, then $s + t \in \mathbb{S}^{\boldsymbol{N}}_{k+1}$.*

Proof. Let $\mathcal{A}, \mathcal{A}' \in k\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}$ computing respectively s and t. Suppose that

$$q_0 a_{k+1}[T_{k,2}[a_1{}^n]]q_0 \vdash^*_{\mathcal{A}} \alpha^{s(n)} \text{ and } q_0 b_{k+1}[T_{k,2}[a_1{}^n]]q_0 \vdash^*_{\mathcal{A}'} \alpha^{t(n)}.$$

To compute $s+t$, it suffices to construct $\mathcal{B} \in k\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}$ producing the following computation: starting from the total state $q_0 c_{k+1}[T_{k,2}[a_1{}^n]]$ (where c_{k+1} is a new symbol), by applying the instruction $\text{change}_{b_{k+1}}$ followed with $\text{push}_{a_{k+1}}$, we obtain the total state $q_0 a_{k+1}[T_{k,2}[a_1{}^n]]b_{k+1}[T_{k,2}[a_1{}^n]]$, where q_0 is the starting (and ending) state of automata $\mathcal{A}$ and $\mathcal{A}'$. It mimics then the behaviour of $\mathcal{A}$ on $q_0 a_{k+1}[T_{k,2}[a_1{}^n]]$, then the $\mathcal{A}'$ ones on $q_0 b_{k+1}[T_{k,2}[a_1{}^n]]$, and ends in q_0.

Proposition 7 (Ordinary product). *If $s, t \in \mathbb{S}^{\boldsymbol{N}}_{k+1}$, $k \geq 2$ then $f \odot g \in \mathbb{S}^{\boldsymbol{N}}_{k+1}$.*

Construction: By using Lemma 4, we obtain (after a suitable choice of set of states and pushdown alphabets) $\mathcal{A}, \mathcal{A}' \in (k+1)\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}$, fulfilling conditions:
(**Q1**) $\forall \Omega_k \in \mathcal{I}_k, (q_0, a_{k+1}[a_k[\ldots[a_2[a_1{}^n]]\ldots]\Omega_k], q_0) \vdash^*_{\mathcal{A}} (q_0, \mathrm{A}_{k+1}[\Omega], q_0)^{s(n)}$.
(**Q1′**) $\forall \Omega_k \in \mathcal{I}_k, (q_0, \mathrm{A}_{k+1}[a_k[\ldots[a_2[a_1{}^n]]\ldots]\Omega_k], q_0) \vdash^*_{\mathcal{A}'} (q_0, \mathrm{B}_{k+1}[\Omega], q_0)^{t(n)}$.
(**Q2**) Δ does not contain lefthand side of the form $(q_0, \varepsilon, \varepsilon)$.
(**Q2′**) Δ' does not contain lefthand side of the form $(q_0, \varepsilon, \varepsilon)$.
(**Q3**) Δ does not contain lefthand side of the form $(q_0, \varepsilon, \mathrm{A}_{k+1} \cdot w)$.
(**Q3′**) Δ' does not contain lefthand side of the form $(q_0, \varepsilon, \mathrm{B}_{k+1} \cdot w)$.
(**Q4**) $Q \cap Q' = \{q_0\}$.
(**Q5**) $\forall i \in [1, k]$, $A_i \cap A_i{}' = \{a_i\}$ and $A_{k+1} \cap A_{k+1}{}' = \{\mathrm{A}_{k+1}\}$.
We construct

$$\mathcal{B} = (Q \cup Q', \{\alpha\}, (B_1, \ldots, B_{k+1}), \boldsymbol{N}, \Delta \cup \Delta' \cup \Delta'', q_0, b_{k+1})$$

where $B_i = A_i \cup A'_i$ for $1 \leq i \leq k$ and $B_{k+1} = A_{k+1} \cup A'_{k+1} \cup \{b_{k+1}\}$ et Δ'' is the union for $\boldsymbol{o} \in \{0,1\}^{|\boldsymbol{N}|}$ of the following transitions:

(1) $\Delta''(q_0, \varepsilon, b_{k+1}a_k \cdots a_2, \boldsymbol{o}) = (\text{push}_{a_k}\ \text{change}_{a_{k+1}}, q_0)$
(2) $\Delta''(q_0, \alpha, \mathrm{B}_{k+1}, \boldsymbol{o}) = (\text{pop}_{k+1}, q_0)$

Proof.
Determinism: Consider (q_1, ε, w_1), lefthand side of a rule of Δ, and (q_2, ε, w_2) lefthand side of a rule of Δ'. Each of them can be applied to a same state only if $q_1 = q_2$ and $w_1 = w_2$. In this case, from (**Q4**) $q_1 = q_2 = q_0$ and from (**Q5**) $w_1 = w_2 = \mathrm{A}_{k+1}w$. But the condition (**Q3**) makes impossible such lefthand sides for Δ. Then $\Delta \cup \Delta'$ is deterministic. The addition of transitions (1) does not break the determinism since b_{k+1} is a new symbol, finally from (**Q3'**), a transition (2) is compatible with no transition of Δ', and from (**Q5**) with no transition of Δ. The automaton is then deterministic.

In addition, from (**Q2**) and (**Q2'**), the automaton $\mathcal{B}$ verifies the condition (**H2**) of the Definition 7: there are no transitions whose lefthand side is $(q_0, \varepsilon, \varepsilon)$ or $(q_0, \alpha, \varepsilon)$.
Computation of the sequence: Let us show now that the automaton compute

$f \odot g$. For all $n \geq 0$, the following derivations are valid:

$$\begin{array}{llll}
(q_0 b_{k+1}[T_{k,2}[{a_1}^n]]q_0) & \vdash_{\mathcal{A}} & (q_0 a_{k+1}[T_{k,2}[{a_1}^n]T_{k,2}[{a_1}^n]]q_0) & \text{(by transitions (1))} \\
 & \vdash^*_{\mathcal{A}} & (q_0 \mathrm{A}_{k+1}[T_{k,2}[{a_1}^n]]q_0)^{f(n)} & \text{(by } (\mathbf{Q1})) \\
 & \vdash^*_{\mathcal{A}} & (q_0 \mathrm{B}_{k+1}[\varepsilon]q_0)^{f(n)\cdot g(n)} & \text{(by } (\mathbf{Q1'})) \\
 & \vdash^*_{\mathcal{A}} & \alpha^{f(n)\cdot g(n)} & \text{(by transitions (2))}.
\end{array}$$

Proposition 8. *If $s \in \mathbb{S}^{\boldsymbol{N}}_{k+1}$, $k \geq 2$, then the sequence t defined by $t(0) = c \geq 1$ and $t(n+1) = s(n).t(n)^d$, for $d \geq 1$, belongs to $\mathbb{S}^{\boldsymbol{N}}_{k+1}$.*

Proof. There exists an automaton $\mathcal{A}_1 = (Q, \{\alpha\}, (A_1, \ldots, A_{k+1}), \boldsymbol{N}, \Delta_1, q_0, a_{k+1}) \in k+1\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}$ fulfilling conditions $(\mathbf{Q1}(a_{k+1} \cdots a_1, \mathrm{A}_{k+1})), (\mathbf{Q2}), (\mathbf{Q3}(\mathrm{A}_{k+1}))$ established in Lemma 4. We consider

$$\mathcal{A} = (Q, \{\alpha\}, (B_1, \ldots, B_{k+1}), \boldsymbol{N}, \Delta, q_0, d_{k+1}) \in k\text{-}\mathsf{CPDA}$$

where $B_{k+1} = A_{k+1} \cup \{d_k\}$, $B_i = A_i$ for all $1 \leq i \leq k$ and Δ is union of Δ_1 with the following new transitions:

for all $\boldsymbol{o} \in \{0,1\}^{|\boldsymbol{N}|}$,

(0.1) $\Delta(q_0, \varepsilon, d_{k+1}a_k \cdots a_2 a_1, \boldsymbol{o}) = (\mathrm{pop}_1(\mathrm{push}_{a_k})^d \, \mathrm{change}_{a_{k+1}}, q_0)$,
(0.2) $\Delta(q_0, \varepsilon, d_{k+1}a_k \cdots a_2, \boldsymbol{o}) = (\mathrm{pop}_k(\mathrm{push}_{d_{k+1}})^{c-1}, q_0)$,
(1) $\Delta(q_0, \varepsilon, \mathrm{A}_{k+1}a_k \cdots a_2 a_1, \boldsymbol{o}) = \Delta(q_0, \varepsilon, \mathrm{A}_{k+1}a_k \cdots a_2, \boldsymbol{o}) = (\mathrm{change}_{d_{k+1}}, q_0)$,
(2) $\Delta(q_0, \alpha, d_{k+1}, \boldsymbol{o}) = (\mathrm{pop}_{k+1}, q_0)$.

This automaton is deterministic: Δ_1 is a deterministic and from condition $(\mathbf{Q2})$, transitions (1) do not break the determinism. In addition, transitions (0.i) et (2) are incompatible with all transitions of Δ_1 (since d_{k+1} is a new symbol), and cannot interfer the ones with the others.
In order to show that $\mathcal{A}$ computes t, we enumerate interesting basic derivations:
Initialization rules, (I0):
by using transitions (0.1),

$$(q_0 d_{k+1}[T_{k,2}[{a_1}^{n+1}]\Omega_k]q_0) \vdash^*_{\mathcal{A}} (q_0 a_{k+1}[(T_{k,2}[{a_1}^n])^{d+1}\Omega_k]q_0),$$

Initialization rules, (I0'); by transitions (0.2) and the decomposition rule,

$$(q_0 d_{k+1}[T_{k,2}[\varepsilon]\Omega_k]q_0) \vdash^*_{\mathcal{A}} (q_0 d_{k+1}[\Omega_k]q_0)^c,$$

s-computation rule, (C1):
from $(\mathbf{P1})$,

$$(q_0 a_{k+1}[T_{k,2}[{a_1}^n]\Omega_k]q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{A}_{k+1}[\Omega_k]q_0)^{s(n)},$$

Gluing rule, (R2): by using transitions (1), for all $n \geq 0$

$$(q_0 \mathrm{A}_{k+1}[T_{k,2}[{a_1}^n]\Omega_k]q_0) \vdash^*_{\mathcal{A}} (q_0 d_{k+1}[T_{k,2}[{a_1}^n]\Omega_k]q_0),$$

Ending rule, (T3): by using transitions (2),

$$(q_0 d_{k+1}[\varepsilon]q_0) \vdash_{\mathcal{A}} \alpha.$$

Let us show by induction, the following property $\mathbf{P}(n)$:

$$(q_0 d_{k+1}[T_{k,2}[a_1{}^n]\Omega_k]q_0)\vdash_{\mathcal{A}}^*(q_0 d_{k+1}[\Omega_k]q_0)^{t(n)}.$$

<u>*Basis*</u>: The initialization rule (I0') proves $\mathbf{P}(0)$.
<u>*Induction step*</u>: Consider the derivation:

$$\begin{aligned}(q_0 d_{k+1}[T_{k,2}[a_1{}^{n+1}]\Omega_k]q_0)\vdash_{\mathcal{A}}^* & \ (q_0 a_{k+1}[(T_{k,2}[a_1{}^n])^{d+1}\Omega_k]q_0) \text{ (by rule (I0))},\\ & \vdash_{\mathcal{A}}^* (q_0 \mathrm{A}_{k+1}[(T_{k,2}[a_1{}^n])^d\Omega_k]q_0)^{s(n)} \text{ (by } (C1)),\\ & \vdash_{\mathcal{A}}^* (q_0 d_{k+1}[(T_{k,2}[a_1{}^n])^d\Omega_k]q_0)^{s(n)} \text{ (by rule (R2))} \quad (6)\end{aligned}$$

By applying d times $\mathbf{P}(n)$ (with suitable substitutions of the indeterminate Ω_k), we obtain:

$$(q_0 d_{k+1}[(T_{k,2}[a_1{}^n])^d\Omega_k]q_0)\vdash_{\mathcal{A}}^*(q_0 d_{k+1}[\Omega_k]q_0)^{t(n)^d} \quad (7)$$

The composition of derivations (6) and (7) gives:

$$(q_0 d_{k+1}[T_{k,2}[a_1{}^{n+1}]\Omega_k]q_0)\vdash_{\mathcal{A}}^*(q_0 d_{k+1}[\Omega_k]q_0)^{s(n)\cdot t(n)^d} = (q_0 d_{k+1}[\Omega_k]q_0)^{t(n+1)}$$

The property $\mathbf{P}(n+1)$ is then true, then $\mathbf{P}(n)$ is true for all $n \geq 0$.
By applying the ending rule (T3) to $\mathbf{P}(n)$, we deduce that, for all $n \geq 0$,

$$(q_0 d_{k+1}[T_{k,2}[a_1{}^n]]q_0)\vdash_{\mathcal{A}}^*\alpha^{t(n)}$$

Let us notice that, by Propositions 6 and 7, for all $k \geq 3$, $(\mathbb{S}_k^{\boldsymbol{N}}, +, \cdot)$ is a semi-ring. We denote by $P(n, X_1, \ldots, X_p)$ any element of the semi-ring $\mathbb{S}_k[X_1, \ldots, X_p]$ to emphasise the fact that coefficients of P are functions of the integer argument n.

Proposition 9. *Let $k \geq 2$, let $P_i(n, X_1, \ldots, X_p)$, $1 \leq i \leq p$ be polynomials with coefficients in $\mathbb{S}_{k+1}^{\boldsymbol{N}}$ and u_i, for $1 \leq i \leq p$, be sequences defined by $u_i(n+1) = P_i(n, u_1(n), \ldots, u_p(n))$, and $u_i(0) = \gamma_i$. Then $u_1 \in \mathbb{S}_{k+1}^{\boldsymbol{N}}$.*

Sketch of proof: The principle used here is the same as the one exposed in proof of Proposition 8, extended to several indeterminates. Suppose that for all $1 \leq i \leq p$,

$$P_i(n, X_1, \ldots X_p) = \sum_{j=0}^{\nu_i} u_{i,j}(n) X_1^{d_{i,j,1}} \ldots X_p^{d_{i,j,p}}.$$

By using Lemma 4, we can suppose that each coefficient $u_{i,j}(n)$ is computed by an automaton,

$$\mathcal{A}_{i,j} = (Q_{i,j}, \emptyset, (A_{1,i,j}, \ldots, A_{k,i,j}), \boldsymbol{N}, \Delta_{i,j}, q_0, a_{k+1,i,j}) \in (k+1)\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}$$

fulfilling conditions $(\mathbf{Q1}(a_{k+1,i,j}, a_k \cdots a_1, \mathrm{A}_{k+1,i,j}))$, $(\mathbf{Q2})$ and $(\mathbf{Q3}(\mathrm{A}_{k+1,i,j}))$ defined in Lemma 4. By a renaming of states and pushdown alphabets, we obtain for all couple $(i,j) \neq (i',j')$:

$Q_{i,j} \cap Q_{i',j'} = \{q_0\}$; $\forall l \in [1,k]$, $A_{l,i,j} \cap A_{l,i',j'} = \{a_i\}$ and $A_{k+1,i,j} \cap A_{k+1,i',j'} = \emptyset$.

Given $\mathcal{A} = (Q, \{\alpha\}, (A_1, \ldots, A_{k+1}), \boldsymbol{N}, \Delta, q_0, a_{k+1}) \in (k+1)\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}$ such that for all (i,j), Q contains all $Q_{i,j}$, Δ contains all $\Delta_{i,j}$, and for all ℓ, A_ℓ contains all $A_{l,i,j}$. Suppose in addition that A_k contains the new symbols $\mathrm{U}_{k,1}, \mathrm{U}_{k,2}, \ldots, \mathrm{U}_{k,p}$ and A_{k+1} contains the new symbol a_{k+1}. Suppose that transitions allow the following basic derivations:
initialization rules:

$$(q_0 a_{k+1}[\mathrm{U}_{k,i}[T_{k-1,2}[{a_1}^{n+1}]]\Omega_k]q_0) \vdash^*_{\mathcal{A}} \prod_{j=0}^{\nu_i} (q_0 a_{k+1,i,j}[(T_{k,2}[{a_1}^{n}])^2 \Omega_k]q_0)$$

and

$$(q_0 a_{k+1}[\mathrm{U}_{k,i}[T_{k-1,2}[\varepsilon]]\Omega]q_0) \vdash^*_{\mathcal{A}} (q_0 a_{k+1}[\Omega_k]q_0)^{\gamma_i}$$

coefficients rules:

$$(q_0 a_{k+1,i,j}[T_{k,2}[{a_1}^{n}]\Omega_k]q_0) \vdash^*_{\mathcal{A}_{i,j}} (q_0 \mathrm{A}_{k+1,i,j}[\Omega_k]q_0)^{u_{i,j}(n)}$$

(it is just the condition $(\mathbf{Q1})$ for $\mathcal{A}_{i,j}$)
gluing rules: for all $n \geq 0$, the gluing rule is:

$$(q_0 \mathrm{A}_{k+1,i,j}[T_{k,2}[{a_1}^{n}]\Omega_k]q_0) \vdash^*_{\mathcal{A}} (q_0 a_{k+1}[(\prod_{\ell=1}^{p} (\mathrm{U}_{k,\ell}[T_{k-1,2}[{a_1}^{n}]])^{d_{i,j,\ell}})\Omega_k]q_0)$$

termination rule:

$$(q_0 a_{k+1}[\varepsilon]q_0) \vdash_{\mathcal{A}} \alpha$$

Consider the property $\mathbf{P}(n)$ defined by:

$$\forall i \in [1,p],\ (q_0 a_{k+1}[\mathrm{U}_{k,i}[T_{k-1,2}[{a_1}^{n}]]\Omega_k]q_0) \vdash^*_{\mathcal{A}} (q_0 a_{k+1}[\Omega_k]q_0)^{u_i(n)}.$$

The property $\mathbf{P(n)}$ can easily be verified by an induction over n: by applying successively initialization, coefficients and gluing rules, we obtain the following derivation:

$$(q_0 a_{k+1}[\mathrm{U}_{k,i}[T_{k-1,2}[{a_1}^{n+1}]]\Omega_k]q_0) \vdash^*_{\mathcal{A}} \prod_{j=0}^{\nu_i} (q_0 a_{k+1}[(\prod_{\ell=1}^{p} (\mathrm{U}_{k,\ell}[T_{k-1,2}[{a_1}^{n}]])^{d_{i,j,\ell}})\Omega_k]q_0)^{u_{i,j}(n)}$$

By applying hypothesis $\mathbf{P}(n)$, we obtain $\mathbf{P}(n+1)$. Applying the termination rule to $\mathbf{P}(n)$, we prove that this automaton computes the sequence u_i.

Using the normalization properties $(\mathbf{Q2})$ et $(\mathbf{Q3})$, it is possible to add transitions to the union of $\Delta_{i,j}$, in a such way as all these rules are valid and $\mathcal{A}$ stays deterministic:

- variables of lefthand sides of the rules above are different,
- it suffices then to decompose each rule in a finite sequence of elementary steps, using some disjoint sets of states for intermediary transitions, to obtain a such a deterministic automaton.

Proposition 10. *Let $f \in \mathbb{S}_{k+1}^{\boldsymbol{N}}, g \in \mathbb{S}_k{}^{\boldsymbol{N}}$, $k \geq 3$. Then the sequence h defined for all $n \geq 0$ by $h(n) = f(n)^{g(n)}$ belongs to $\mathbb{S}_{k+1}^{\boldsymbol{N}}$.*

Proof. Let us proceed as in the proof of Proposition 4: we expose, in a first step, a list of particular derivations (that we call "rules") and prove that these rules are sufficient to compute the required sequence; in a second step, we explain how to construct a deterministic automaton which makes these rules available..
First step:
Let $\mathcal{A} = (Q, \{\alpha\}, (A_1, \ldots, A_{k+1}), \boldsymbol{N}, \Delta, q_0, a_{k+1}) \in k+1\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}$ with $A_{k+1} \supseteq \{a_{k+1}, \mathrm{A}_{k+1}, b_{k+1}\}$, and $A_k \supseteq \{a_k, b_k, \mathrm{B}_k\}$ and for all $i \in [1, k-1]$, $A_i \supseteq \{a_i\}$.
Let us define $a_{k+1}{}^{-1} \cdot \mathcal{A} = (Q, \{\alpha\}, (A_1, \ldots, A_k), \boldsymbol{N}, a_{k+1}{}^{-1} \cdot \Delta, q_0, a_k) \in k\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}$, where

$$a_{k+1}{}^{-1} \cdot \Delta = \{\delta \mid \delta = (q, \alpha_\varepsilon, a_{k+1}w, \boldsymbol{o}, q', \mathrm{instr}) \in \Delta \text{ and } \mathrm{instr} \in \mathcal{I}nstr_k\}.$$

We suppose that $\mathcal{A}$ allows the following basic derivations (where Ω_i is an indeterminate of level i):
initialization rule, (I0):

$$(q_0 b_{k+1}[T_{k,2}[a_1{}^n]]q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{A}_{k+1}[b_k[T_{k-1,2}[a_1{}^n]\ T_{k-1,2}[a_1{}^n]]]q_0),$$

f-computation, (C1):

$$(q_0 a_{k+1}[T_{k,2}[a_1{}^n]\Omega_k]q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{A}_{k+1}[\Omega_k]q_0)^{f(n)},$$

g-computation, (C2):

$$(r_0 b_k[T_{k-1,2}[a_1{}^n]\Omega_{k-1}]r_0) \vdash^*_{a_{k+1}{}^{-1}\cdot\mathcal{A}} (r_0 \mathrm{B}_k[\Omega_{k-1}]r_0)^{g(n)},$$

gluing rules, (R12): $\forall \omega_k \in k\text{-}\mathsf{pds}$,

$$(q_0 \mathrm{A}_{k+1}[\omega_k]q_0) \vdash_{\mathcal{A}} (r_0 a_{k+1}[\omega_k]q_0),$$

gluing rules, (R21): $\forall \omega \in (k-1)\text{-}\mathsf{pds}$,

$$(r_0 a_{k+1}[\mathrm{B}_k[\omega_{k-1}]\Omega_k]q_0) \vdash_{\mathcal{A}} (q_0 a_{k+1}[a_k[\omega_{k-1}]\Omega_k]q_0),$$

gluing rules, (R$^{(0)}$21):

$$(r_0 a_{k+1}[\varepsilon]q_0) \vdash_{\mathcal{A}} (q_0 \mathrm{A}_{k+1}[\varepsilon]q_0),$$

ending rules, (T3):

$$(q_0 \mathrm{A}_{k+1}[\varepsilon]q_0) \vdash_{\mathcal{A}} \alpha.$$

The intuition behind these rules is that gluing rules (Rij) allow to connect the end of a computation (Ci) with the beginning of a computation (Cj). The special gluing rule ($\mathrm{R}^{(0)}21$) handles the case where the computation (C2) results in the number 0, leading to the value $f(n)^0 = 1$. [1]
Let us prove by induction over $i \geq 0$ the following property $\mathbf{P}(i)$:
for every $\omega_i \in k\text{-}\mathsf{pds}(A_1, \ldots, A_k)$, if

$$(r_0 \omega_i r_0) \vdash^*_{a_{k+1}{}^{-1}\mathcal{A}} (r_0 \mathrm{B}_k[T_{k-1,2}[a_1{}^n]] r_0)^i \tag{8}$$

then

$$(q_0 \mathrm{A}_{k+1}[\omega_i] q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{A}_{k+1}[\varepsilon] q_0)^{f(n)^i}. \tag{9}$$

<u>*Basis:*</u> $i = 0$
We suppose that (8) holds. The following derivation is then valid:

$$\begin{array}{ll} & (q_0 \mathrm{A}_{k+1}[\omega_0] q_0) \\ \vdash_{\mathcal{A}} & (r_0 a_{k+1}[\omega_0] q_0) \text{ (by rule (R12))} \\ \vdash^*_{\mathcal{A}} & (r_0 a_{k+1}[\varepsilon] q_0) \text{ (by hypothesis (8) and definition of } a_{k+1}{}^{-1} \cdot \mathcal{A}) \\ \vdash^*_{\mathcal{A}} & (q_0 \mathrm{A}_{k+1}[\varepsilon] q_0) \text{ (by rule } (\mathrm{R}^{(0)}21)). \end{array}$$

<u>*Induction step*</u>:
We suppose that hypothesis (8) is fulfilled by $i+1$ and that $\mathbf{P}(i)$ holds. By means of Lemma 2, we can translate hypothesis (8) into:
there exists $\omega_i \in k\text{-}\mathsf{pds}(A_1, \ldots, A_k)$ such that

$$(r_0, \varepsilon, \omega_{i+1}) \overset{*}{\to}_{a_{k+1}{}^{-1}\cdot\mathcal{A}} (r_0, \varepsilon, \mathrm{B}_k[T_{k-1,2}[a_1{}^n]]\omega_i) \text{ and } (r_0 \omega_i r_0) \vdash^*_{a_{k+1}{}^{-1}\cdot\mathcal{A}} (r_0 \mathrm{B}_k[T_{k-1,2}[a_1{}^n]] r_0)^i.$$

We obtain the derivation:

$$\begin{array}{rcl} (q_0 \mathrm{A}_{k+1}[\omega_{i+1}] q_0) \vdash^*_{\mathcal{A}} & (r_0 a_{k+1}[\omega_{i+1}] q_0) & \text{(by rule (R12))} \\ \vdash^*_{\mathcal{A}} & (r_0 a_{k+1}[\mathrm{B}_k[T_{k-1,2}[a_1{}^n]]\omega_i] q_0) & \text{(by above translation)} \\ \vdash^*_{\mathcal{A}} & (q_0 a_{k+1}[a_k[T_{k-1,2}[a_1{}^n]]\omega_i] q_0) & \text{(by rule (R21))} \\ \vdash^*_{\mathcal{A}} & (q_0 \mathrm{A}_{k+1}[\omega_i] q_0)^{f(n)} & \text{(by (C1)).} \end{array} \tag{10}$$

Combining this derivation with $\mathbf{P}(i)$, we get:

$$(q_0 \mathrm{A}_{k+1}[\omega_{i+1}] q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{A}_{k+1}[\varepsilon] q_0)^{f(n)^{i+1}}.$$

(end of induction).
Let us consider $\omega = b_k[T_{k-1,2}[a_1{}^n] T_{k-1,2}[a_1{}^n]]$. By rule (C2), ω fulfills hypothesis (8) for the integer $i = g(n)$. Hence, by $\mathbf{P}(i)$,

$$(q_0 \mathrm{A}_{k+1}[b_k[T_{k-1,2}[a_1{}^n] T_{k-1,2}[a_1{}^n]]] q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{A}_{k+1}[\varepsilon] q_0)^{f(n)^{g(n)}}. \tag{11}$$

Finally, by applying the initialization rule, the derivation (11), and the ending rule (T3), we get

$$(q_0, b_{k+1}[T_{k,2}[a_1{}^n]], q_0) \vdash^*_{\mathcal{A}} \alpha^{f(n)^{g(n)}}.$$

[1] we adopt the convention that $0^0 = 1$ in the definition of $h = f^g$.

Second step

Let us construct such an automaton. We suppose that the sequence $f(n)$ is computed $\mathcal{A}_1 = (Q_1, \emptyset, (B_1, \dots, B_{k+1}), \boldsymbol{N}, \Delta_1, q_0, a_{k+1}) \in k+1\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}$ fulfilling conditions $(\mathbf{Q1}(a_{k+1}, \cdots a_1, \mathrm{A}_{k+1})), (\mathbf{Q2})$ and $(\mathbf{Q3}(\mathrm{A}_{k+1})$ stated in Lemma 4. As well, the sequence $g(n)$ is computed by $\mathcal{A}_2 = (Q_2, \emptyset, (C_1, \dots, C_k), \boldsymbol{N}, \Delta_2, r_0, b_k) \in k\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}$ fulfilling the same conditions (for symbols $b_k, a_{k-1}, \dots, a_1$ and the ending symbol B_k).

We suppose that $Q_1 \cap Q_2 = \emptyset$, $B_k \cap C_k = \emptyset$ and for $i \in [1, k-1]$, $B_i \cap C_i = \{a_i\}$. Let us define $\mathcal{A} = (Q, \{\alpha\}, (A_1, \dots, A_{k+1}), \boldsymbol{N}, \Delta, q_0, b_{k+1})$ where

$$Q = Q_1 \cup Q_2, \quad A_i = B_1 \cup C_i \text{ for } i \in [1, k], \quad A_{k+1} = B_{k+1} \cup \{b_{k+1}\},$$

and Δ is the union of $\Delta_1 \cup (a_{k+1} \cdot \Delta_2)$ with the following rules:

(0) $\Delta(q_0, \varepsilon, b_{k+1} a_k \cdots a_2) = \Delta(q_0, \varepsilon, b_{k+1} a_k \cdots a_2 a_1) =$ $(\mathrm{change}_{b_k}\ \mathrm{push}_{a_{k-1}}\ \mathrm{change}_{\mathrm{A}_{k+1}}, r_0)$,
(1.2) $\Delta(q_0, \varepsilon, \mathrm{A}_{k+1} w) = (\mathrm{change}_{a_{k+1}}, r_0)$ for all $w \neq \varepsilon \in \mathrm{top}(k\text{-}\mathsf{pds}(\mathcal{A}_k))$,
(2.1.0) $\Delta(r_0, \varepsilon, a_{k+1}) = (\mathrm{change}_{\mathrm{A}_{k+1}}, q_0)$,
(2.1) $\Delta(r_0, \varepsilon, a_{k+1} \mathrm{B}_k w) = (\mathrm{change}_{a_k}, q_0)$ for all $w \in \mathrm{top}((k-1)\text{-}\mathsf{pds}(\mathcal{A}_{k-1}))$,
(3) $\Delta(q_0, \alpha, \mathrm{A}_{k+1}) = (q_0, \mathrm{pop}_{k+1})$.

Since $\mathcal{A}_1$ fulfills $(\mathbf{Q3}(\mathrm{A}_{k+1}))$, transitions (1.2) and (3) do not introduce any non-determinism, as well, for transitions (2.1), since $\mathcal{A}_2$ fulfills $(\mathbf{Q3}(\mathrm{B}_k))$ and $\mathrm{B}_k \notin B_k$. Transitions (2.1.0) use the pair (r_0, a_{k+1}) which is not used in $a_{k+1} \cdot \Delta_2$ (by hypothesis $(\mathbf{Q2})$). Transitions (0) use a new symbol. Then $\mathcal{A}$ is deterministic and fulfills $(\mathbf{H2})$.
The transitions are chosen so as to make the rules (describes in first step) available: (C1) holds by the choice of Δ_1, (C2) holds by the choice of $a_{k+1} \cdot \Delta_2$, (R21) holds by transitions (2.1), ($\mathrm{R}^{(0)}21$) holds by transition (2.1.0), (R12) hold by transitions (1.2) and (T3) holds by transition (3).

Proposition 11. *Let $k \geq 3$. Let $P_i(n, X_1, \dots, X_p)$, $1 \leq i \leq p$ be polynomial with coefficients in $\mathbb{S}^{\boldsymbol{N}}_{k+1}$ and exponents in $\mathbb{S}^{\boldsymbol{N}}_k$. We consider sequences u_i, for $1 \leq i \leq p$ defined by*
$u_i(n+1) = P_i(n, u_1(n), \dots, u_p(n))$, and $u_i(0) = c_i$. Then $u_1 \in \mathbb{S}^{\boldsymbol{N}}_{k+1}$.

Sketch of proof: The principle used here is the same as the one exposed in proof of Proposition 10, extended to several indeterminates. . We suppose that for all $1 \leq i \leq p$,

$$P_i(n, X_1, \dots X_p) = \sum_{j=0}^{\nu_i} u_{i,j}(n) X_1^{d_{i,j,1}(n)} \dots X_p^{d_{i,j,p}(n)}.$$

By using Lemma 4, we can suppose that:

– each coefficient $u_{i,j}(n)$ is computed by an automaton

$$\mathcal{A}_{i,j} = (Q_{i,j}, \emptyset, (A_{1,i,j}, \dots, A_{k,i,j}), \boldsymbol{N}, \Delta_{i,j}, q_0, a_{k+1,i,j}) \in (k+1)\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}$$

fulfilling conditions $(\mathbf{Q1}(a_{k+1,i,j} a_k \cdots a_1))$, $(\mathbf{Q2})$ and $(\mathbf{Q3}(\mathrm{A}_{k+1,i,j}))$ defined in Lemma 4.

– each coefficient $d_{i,j,\ell}(n)$ is computed by an automaton

$$\mathcal{B}_{i,j,\ell} = (Q_{i,j,\ell}, \emptyset, (B_{1,i,j,\ell}, \ldots, B_{k,i,j,\ell}), \boldsymbol{N}, \Delta_{i,j,\ell}, r_0, b_{k,i,j,\ell}) \in k\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}$$

fulfilling conditions ($\mathbf{Q1}(b_{k,i,j,\ell} a_{k-1} \cdots a_1)$), ($\mathbf{Q2}$) and ($\mathbf{Q3}(\mathrm{B}_{k,i,j,\ell})$) defined in Lemma 4.

By a renaming of states and pushdown alphabets, we obtain for all couple $(i,j) \neq (i',j')$:

- $Q_{i,j} \cap Q_{i',j'} = \{q_0\}$,
- $\forall m \in [1,k]$, $A_{m,i,j} \cap A_{m,i',j'} = \{a_i\}$ and
- $A_{k+1,i,j} \cap A_{k+1,i',j'} = \emptyset$,

and for all $(i,j,\ell) \neq (i',j',\ell')$:

- $Q_{i,j,\ell} \cap Q_{i',j',\ell'} = \{r_0\}$,
- $\forall m \in [1,k-1]$, $B_{m,i,j,\ell} \cap B_{m,i',j',\ell'} = \{a_i\}$ and
- $B_{k+1,i,j,\ell} \cap B_{k+1,i',j',\ell'} = \emptyset$,

finally for all (i,j), (i',j',ℓ'):

$Q_{i,j} \cap Q_{i',j',\ell'} = \emptyset$, $\forall m \in [1,k]$, $A_{m,i,j} \cap B_{m,i',j',\ell'} = \emptyset$.

We suppose we are given $\mathcal{A} = (Q, \{\alpha\}, \mathcal{A}_{k+1}, \boldsymbol{N}, \Delta, q_0, a_{k+1}) \in (k+1)\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}$ such that for all (i,j,ℓ), Q contains all $Q_{i,j} \cup Q_{i,j,\ell}$, for all $m \in [1,k]$, A_m contains all $A_{m,i,j} \cup B_{m,i,j,\ell}$, A_{k+1} contains all $A_{k+1,i,j}$ and Δ contains all $\Delta_{i,j} \cup \Delta_{i,j,\ell}$. We suppose in addition that A_k contains the new symbols $\mathrm{U}_{k,1}, \mathrm{U}_{k,2}, \ldots, \mathrm{U}_{k,p}$ and A_{k+1} contains the new symbol a_{k+1}. We suppose finally that transitions allow the following basic derivations:

initialization rules:

$$(q_0 a_{k+1}[\mathrm{U}_{k,i}[T_{k-1,2}[{a_1}^{n+1}]]\Omega_k]q_0) \vdash^*_{\mathcal{A}} \prod_{j=1}^{\nu_i} (q_0 a_{k+1,i,j}[T_{k,2}[{a_1}^n][T_{k,2}[{a_1}^n]]\Omega_k]q_0),$$

coefficients rules:

$$(q_0 a_{k+1,i,j}[T_{k,2}[{a_1}^n]\Omega_k]q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{A}_{k+1,i,j}[\Omega_k]q_0)^{u_{i,j}(n)},$$

(it is just the condition ($\mathbf{Q1}$) for the automaton $\mathcal{A}_{i,j}$)

gluing rules, (R1):

$$(q_0 \mathrm{A}_{k+1,i,j}[T_{k,2}[{a_1}^n]\Omega_k]q_0) \vdash^*_{\mathcal{A}} (q_0 a_{k+1}[(\prod_{\ell=1}^{p} b_{k,i,j,\ell}[T_{k-1,2}[{a_1}^n]T_{k-1,2}[{a_1}^n]])\Omega_k]r_0),$$

degrees rules:

$$(r_0 b_{k,i,j,\ell}[T_{k-1,2}[{a_1}^n]\Omega_{k-1}]r_0) \vdash^*_{{a_{k+1}}^{-1} \cdot \mathcal{A}} (r_0 \mathrm{B}_{k,i,j,\ell}[\Omega_{k-1}]r_0)^{d_{i,j,\ell}(n)},$$

gluing rules, (R2):

$$(r_0 a_{k+1}[\mathrm{B}_{k,i,j,\ell}[T_{k-1,2}[{a_1}^n]\Omega_k]q_0) \vdash^*_{\mathcal{A}} (q_0 a_{k+1}[\mathrm{U}_{k,\ell}[T_{k-1,2}[{a_1}^n]]\Omega_k]q_0),$$

ending rule:

$$(q_0 a_{k+1}[\varepsilon]q_0) \vdash_{\mathcal{A}} \alpha.$$

We consider the property $\mathbf{P}(n)$ defined by:

$$\forall i \in [1,p], (q_0 a_{k+1}[\mathrm{U}_{k,i}[T_{k-1,2}[{a_1}^n]]\Omega]q_0) \vdash_{\mathcal{A}}^{*} (q_0 a_{k+1}[\Omega]q_0)^{u_i(n)}.$$

The property $\mathbf{P}(n)$ can be proved by induction over n: by applying initialization rule, then coefficients rule, then gluing rule (R1), then degrees rule, and finally gluing (R2), we get the following derivation:

$$(q_0 a_{k+1}[\mathrm{U}_{k,i}[T_{k-1,2}[{a_1}^{n+1}]]\Omega_k]q_0) \vdash_{\mathcal{A}}^{*} \prod_{j=0}^{\nu_i}(q_0 a_{k+1}[(\prod_{\ell=1}^{p}(\mathrm{U}_{k,\ell}[T_{k-1,2}[{a_1}^n]])^{d_{i,j,\ell}(n)})\Omega_k]q_0)^{u_{i,j}(n)}$$

From hypothesis $\mathbf{P}(n)$, we obtain $\mathbf{P}(n+1)$. The application of the ending rule to $\mathbf{P}(n)$ prove then that this automaton computes the sequence u_i.

Using the normalization properties (**Q2**) et (**Q3**), it is possible to add transitions to the union of $\Delta_{i,j}$, in a such way as all these rules are valid and $\mathcal{A}$ stays deterministic:
- variables of lefthand sides of the rules above are different,
- it suffices then to decompose each rule in a finite sequence of elementary steps, using some disjoint sets of states for intermediary transitions, to obtain a such a deterministic automaton.

Proposition 12 (Convolution-product). *Let $f \in \mathbb{S}_{k+1}^{\boldsymbol{N}}$ and $g \in \mathbb{S}_k$, for $k \geq 3$. Then $f \times g \in \mathbb{S}_{k+1}^{\boldsymbol{N}}$ where $f \times g$ denotes the convolution-product:*

$$(f \times g)(n) = \sum_{m=0}^{n} f(m) \cdot g(n-m) \text{ for all } n \in \mathbb{N}$$

Proof. The major difficulty is to define for all $0 \leq m \leq n$, a $(k+1)$-pds $\omega_{m,n-m}$ from which we can calculate $f(m) \cdot g(n-m)$, then to generate the sequence $\omega_{n,0}$, $\omega_{n-1,1}, \ldots, \omega_{0,n}$. This time, we need two counters which must be able to evolve simultaneously, it is then not possible to place them both at level 1. The counter a_1 of the sequence f will be placed at level 1 as usually, while that of sequence g, denoted b_2 will be placed at level 2.

That is why sequences f and g belong to two classes of different levels and why we do not authorize the automaton computing g to be controlled. We code each couple $(m, n-m)$ by the following 2-pds:

$$\gamma_{m,n} = a_2[a_1^m]b_2[a_1^{m+1}]\cdots b_2[{a_1}^n],\ m \neq n,\ \gamma_{n,n} = a_2[{a_1}^n]$$

The integer m is coded as usually in the first atom, while the integer $n-m$ corresponds to the length of the suffix $b_2[a_1^{m+1}]\cdots b_2[{a_1}^n]$. We compute then $f(m) \cdot g(n-m)$ by using the same kind of argument as in Proposition 7 concerning the product.

First step

Let us suppose we are given $\mathcal{A} = (Q, \{\alpha\}, \mathcal{A}_{k+1}, \boldsymbol{N}, \Delta, q_0) \in (k+1)\text{-}\mathsf{CPDA}^{\boldsymbol{N}}$ with $A_1 = \{a_1\}$, $A_2 \supseteq \{a_2, b_2\}$, $A_i \supseteq \{a_i\}$ for $i \in [3,k]$ et $A_{k+1} \supseteq \{a_{k+1}, \mathrm{A}_{k+1}\mathrm{B}_{k+1}\}$.

As previously, the letter Ω_i is an indeterminate of level i. Letters a_2 and b_2 are used as "counters" for the sequence g, while a_1 is the counter used for the sequence f. We suppose that $\mathcal{A}$ allows the following basic derivations:
f-computation, (C1): for all $n, m \geq 0$,

$$(q_0 b_{k+1}[T_{k,3}[\gamma_{m,n}]\Omega_k]q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{B}_{k+1}[\Omega_k]q_0)^{f(m)},$$

g-computation, (C2): for all $n \geq 0, m \in [0,n]$,

$$(q_0 \mathrm{B}_{k+1}[T_{k,3}[\gamma_{n,m}]]q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{A}_{k+1}[\varepsilon]q_0)^{g(m)},$$

pair-generation, (G3): for all $1 \leq m \leq n$,

$$(q_0 a_{k+1}[T_{k,3}[\gamma_{m,n}]]q_0) \vdash^*_{\mathcal{A}} (q_0 a_{k+1}[T_{k,3}[\gamma_{m-1,n}]]q_0)(q_0 b_{k+1}[T_{k,3}[\gamma_{m,n}]T_{k,3}[\gamma_{m,n}]]q_0),$$

initial pair generation, (G30): for all $n \geq 0$,

$$(q_0 a_{k+1}[T_{k,3}[\gamma_{0,n}]]q_0) \vdash^*_{\mathcal{A}} (q_0 b_{k+1}[T_{k,3}[\gamma_{0,n}]T_{k,3}[\gamma_{0,n}]]q_0),$$

ending rule, (T4):

$$(q_0 \mathrm{A}_{k+1}[\varepsilon]q_0) \vdash^*_{\mathcal{A}} \alpha.$$

Since $\gamma_{n,n} = a_2[{a_1}^n]$, by applying iteratively (G3) then (G30), we get:

$$(q_0 a_{k+1}[T_{k,2}[{a_1}^n]]q_0) \vdash^*_{\mathcal{A}} \prod_{m=0}^{n} (q_0 b_{k+1}[T_{k,3}[\gamma_{m,n}]T_{k,3}[\gamma_{m,n}]]q_0). \tag{12}$$

Starting with each factor of this product, we derive:

$$\begin{aligned}(q_0 b_{k+1}[T_{k,3}[\gamma_{m,n}]T_{k,3}[\gamma_{m,n}]]q_0) &\vdash^*_{\mathcal{A}} (q_0 \mathrm{B}_{k+1}[T_{k,3}[\gamma_{m,n}]]q_0)^{f(m)} \text{ (by (C1))}\\ &\vdash^*_{\mathcal{A}} (q_0 \mathrm{A}_{k+1}[\varepsilon]q_0)^{g(n-m)\cdot f(m)} \text{ (by (C2))}.\end{aligned} \tag{13}$$

Combining the two derivations (12) and (13), we get:

$$\begin{aligned}(q_0 a_{k+1}[T_{k,3}[a_2[{a_1}^n]]]q_0) &\vdash^*_{\mathcal{A}} (q_0 \mathrm{A}_{k+1}[\varepsilon]q_0)^{\sum_{m=0}^{n} g(n-m)\cdot f(m)}\\ &= (q_0 \mathrm{A}_{k+1}[\varepsilon]q_0)^{(f \times g)(n)}.\end{aligned}$$

Second step:
Let us construct a such an automaton $\mathcal{A}$. The sequence $f(n)$ is computed by some $(k+1)$-ACD $\mathcal{A}'_1 \in k+1\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}$ fulfilling conditions $(\mathbf{Q1}(b_{k+1}a_k \cdots a_1, \mathrm{B}_{k+1})$, $(\mathbf{Q2})$ and $(\mathbf{Q3}(\mathrm{B}_{k+1}))$ states in Lemma 4. We introduce a new symbol b_2, by adding transitions allowing to see b_2 as a a bottom symbol. We obtain then an automaton $\mathcal{A}_1 = (Q_1, \emptyset, (B_1, \ldots, B_{k+1}), \boldsymbol{N}, \Delta_1, q_0, b_{k+1})$, where $B_{k+1} \supseteq \{b_{k+1}, \mathrm{B}_{k+1}\}$, $B_i \supseteq \{a_i\}$ for $i \in [3,k]$, $B_2 \supseteq \{a_2, b_2\}$ et $B_1 = \{a_1\}$ fulfilling conditions (C1), $(\mathbf{Q2})$ and $(\mathbf{Q3}(\mathrm{B}_{k+1}))$.

From Lemma 3, the sequence $g(n)$ is computed by an automaton

$$\mathcal{A}'_2 = (Q'_2, \emptyset, (C_2, \ldots, C_{k+1}), \Delta'_2, q_0) \in k\text{-DCPDA}$$

where $C_2 = \{b_2\}$, $C_i \supseteq \{a_i\}$ for $i \in [3, k]$, $C_{k+1} \supseteq \{\mathrm{B}_{k+1}, \mathrm{A}_{k+1}\}$ and fulfilling the condition $(\mathbf{P1}(q_0, \mathrm{B}_{k+1} a_k \cdots a_3 b_2, \mathrm{A}_{k+1}))$, the condition $(\mathbf{P2})$ and the condition $(\mathbf{P3}(\mathrm{A}_{k+1}))$. It is then easy to transform $\mathcal{A}'_2$ into a controlled automaton $\mathcal{A}_2 = (Q_2, \{\alpha\}, (C_1, \ldots, C_{k+1}), \boldsymbol{N}, \Delta_2, q_0) \in k+1\text{-DCPDA}^{\boldsymbol{N}}$, with $C_1 = \{a_1\}$, and fulfilling (C2) and $(\mathbf{P2})$ and $(\mathbf{P3})$:
we pose for each ${\Delta_2}'(p, \varepsilon, w) = (\mathrm{instr}, q)$ such that $|w| = k$,

$$\Delta_2(p, \varepsilon, w, \boldsymbol{o}) = \Delta(p, \varepsilon, wa_1, \boldsymbol{o}) = (\mathrm{instr}^{+1}, q),\ \forall \boldsymbol{o} \in \{0,1\}^{|\boldsymbol{N}|},$$

and for each ${\Delta_2}'(p, \varepsilon, w) = (\mathrm{instr}, q)$ such that $|w| < k$,

$$\Delta_2(p, \varepsilon, w, \boldsymbol{o}) = (\mathrm{instr}^{+1}, q),\ \forall \boldsymbol{o} \in \{0,1\}^{|\boldsymbol{N}|},$$

with for all $i \in [1, k]$, ${\mathrm{pop}_i}^{+1} = pop_{i+1}$, ${\mathrm{push}_a}^{+1} = \mathrm{push}_a$ and ${\mathrm{change}_a}^{+1} = \mathrm{change}_a$.

The symbol a_1 is then ignored by the automaton and controllers have no influence on transitions. We choose the alphabets is a such way as $B_2 \cap C_2 = \{a_2, b_2\}$, $B_i \cap C_i = \{a_i\}$ for $i \in [3, k]$ and $B_{k+1} \cap C_{k+1} = \{\mathrm{B}_{k+1}\}$.
Let us define $\mathcal{A} = (Q, \{\alpha\}, (A_1, \ldots, A_{k+1}), \boldsymbol{N}, \Delta, q_0, a_{k+1})$ where

$$Q = Q_1 \cup Q_2 \cup \{r_1\};\ \ A_i = B_i \cup C_i \text{ for } i \in [1, k], \text{ and } A_{k+1} = B_{k+1} \cup C_{k+1};$$

Δ is the union of $(\Delta_1 \cup \Delta_2)$ with the following transitions: $\forall \boldsymbol{o} \in \{0,1\}^{|\boldsymbol{N}|}$,

(3.1) $\Delta(q_0, \varepsilon, a_{k+1} a_k \cdots a_2 a_1, \boldsymbol{o}) = (\mathrm{push}_{a_k}\, \mathrm{change}_{b_{k+1}}\, \mathrm{push}_{a_{k+1}}\, \mathrm{pop}_k, r_1)$,
(3.2) $\Delta(r_1, \varepsilon, a_{k+1} a_k \cdots a_2 a_1, \boldsymbol{o}) = (\mathrm{change}_{b_2}\, \mathrm{push}_{a_2}\, \mathrm{pop}_1, q_0)$,
(30) $\Delta(q_0, \varepsilon, a_{k+1} a_k \cdots a_2, \boldsymbol{o}) = (\mathrm{push}_{a_k}\, \mathrm{push}_{b_{k+1}}, q_0)$,
(4) $\Delta(q_0, \alpha, \mathrm{A}_{k+1}, \boldsymbol{o}) = (\mathrm{pop}_{k+1}, q_0)$.

Initial automata $\mathcal{A}_i$ are deterministic, and since $\mathcal{A}_2$ fulfills $(\mathbf{P2}(A_{k+1}))$, the new automaton $\mathcal{A}$ is also deterministic.
Transitions are chosen so as to make the rules (described in the first step) available: (Ci) holds by Δ_i, $(i = 1, 2)$, (G3) holds by transitions (3.j), (G30) holds by transitions (30), and (T4) holds by transitions (4). We can then conclude that $\mathcal{A}$ compute $f \times g$.

The generation mode of pairs that we use obliges the final automaton to be of level at least 4 (to obtain $b_{k+1}[T_{k,3}[\gamma_{m,n}] T_{k,3}[\gamma_{m,n}]]$ from $a_{k+1}[T_{k,3}[\gamma_{m,n}]]$). However, the first copy is used to compute $f(m)$ and we thus do not need $T_{k,3}[\gamma_{m,n}]$ entire, but simply of the value of m. Since for the construction of the system allowing derivation (C2), we use the weak version of the lemma of normalization, which is valid for automata of level at least 2, it is in fact possible to prove the proposition for $k = 2$ but the proof is slightly different.

Proposition 13 (Convolution-product). *Let $f \in \mathbb{S}_3{}^{\boldsymbol{N}}$ and $g \in \mathbb{S}_2$. Then $f \times g \in \mathbb{S}_3{}^{\boldsymbol{N}}$ where $f \times g$ denotes the convolution-product:*

$$(f \times g)(n) = \sum_{m=0}^{n} f(m) \cdot g(n-m) \text{ for all } n \in \mathbb{N}.$$

Proof. We use here the same notations as in the proof of the previous lemma, by adding $\gamma'_{m,n} = b_2[{a_1}^{m+1}] \cdots b_2[{a_1}^n]$ (i.e., $\gamma_{m,n} = a_2[{a_1}^m]\gamma'_{m,n}$).
First step: We suppose given a automaton $\mathcal{A} \in k\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}$ for which the following derivations hold:
f-computation, (C1): for all $n \geq 0$,

$$(q_0 b_3[a_2[{a_1}^n]\Omega_2]q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{B}_2[\Omega_2]q_0)^{f(n)},$$

g-computation, (C2): for all $n \geq 0, m \in [0, n]$,

$$(q_0 \mathrm{B}_3[\gamma'_{m,n}]q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{A}_3[\varepsilon]q_0)^{g(n-m)},$$

pair-generation, (G3): for all $1 \leq m \leq n$,

$$(q_0 a_3[\gamma_{m,n}]q_0) \vdash^*_{\mathcal{A}} (q_0 a_3[\gamma_{m-1,n}]q_0)(q_0 b_3[\gamma_{m,n}]q_0),$$

initial pair-generation, (G30): for all $0 \leq n$,

$$(q_0 a_3[\gamma_{0,n}]q_0) \vdash^*_{\mathcal{A}} (q_0 b_3[\gamma_{0,n}]q_0),$$

ending rule, (T4):

$$(q_0 \mathrm{A}_3[\varepsilon]q_0) \vdash^*_{\mathcal{A}} \alpha.$$

From these rules, we obtain the following derivations:

$$\begin{array}{rll}
(q_0 a_3[a_2[{a_1}^n]]q_0) \vdash^*_{\mathcal{A}} & \prod_{m=0}^{n} (q_0 b_3[\gamma_{m,n}]q_0) & \text{(by rules (G3), (G30))} \\
\vdash^*_{\mathcal{A}} & \prod_{m=0}^{n} (q_0 \mathrm{B}_3[\gamma'_{m,n}]q_0)^{f(m)} & \text{(by rule (C1))} \\
\vdash^*_{\mathcal{A}} & \prod_{m=0}^{n} (q_0 \mathrm{A}_3[\varepsilon]q_0) & \text{(by rule (C2))} \\
\vdash^*_{\mathcal{A}} & \alpha^{h(n)} & \text{(by rule (T4)).}
\end{array}$$

Second step:
Lemma 4 allows to obtain transitions making valid (C1), those allowing (C2) are obtained as in the previous proof except that it is not any more necessary to erase $a_2[{a_1}^m]$ before to start the computation. The derivation (T4) is obtain in an obvious way, finally, the system allowing the pair-generation rules is the union for $\boldsymbol{o} \in \{0,1\}^{|\boldsymbol{N}|}$ of transitions:

(3) $\Delta(q_0, \varepsilon, a_3 a_2 a_1, \boldsymbol{o}) = (\mathrm{change}_{b_3}\ \mathrm{push}_{a_3}\ \mathrm{change}_{b_2}\ \mathrm{push}_{a_2}\ \mathrm{pop}_1, q_0)$,
(3.0) $\Delta(q_0, \varepsilon, a_3 a_2, \boldsymbol{o}) = (\mathrm{change}_{b_3}, q_0)$.

With a suitable choice of concrete set of states and pushdown alphabets, we obtain a deterministic automaton computing $f \times g$.

Proposition 14 (Convolution-inverse). *Let $g \in \mathbb{S}_k$, $k \geq 2$, and let f be the sequence defined by: $f(0) = 1$ and for all $n \geq 0$, $f(n+1) = \sum_{m=0}^{n} f(m) \cdot g(n-m)$. Then $f \in \mathbb{S}_{k+1}$.*

Sketch of proof: We follow the same lines as for Proposition 12.
First step: We suppose we are given $\mathcal{A} = (Q, \{\alpha\}, (\mathcal{A}_{k+1}), \Delta, q_0, a_{k+1}) \in k+1$-DCPDA, with $A_{k+1} \supseteq \{a_{k+1}, \mathrm{A}_{k+1}, b_{k+1}, \mathrm{B}_{k+1}\}$, $A_i \supseteq \{a_i\}$ for $i \in [k, 3]$, $A_2 \supseteq \{a_2, b_2, \mathrm{B}_2\}$ and $A_1 = \{a_1\}$, where this new symbol B_2 plays the role of a bottom symbol for the automaton computing g. We call here "blocking pds" every 2-pds ω of the form $\mathrm{B}_2[\omega_1] \cdot \omega_2$, for $\omega_i \in i$-pds or $\omega = \varepsilon$. As previously, we encode the pair $(m, n-m)$ by the 2-pds

$$\gamma_{m,n} = a_2[a_1^m] b_2[a_1^{m+1}] \cdots b_2[{a_1}^n].$$

Blocking pds will be used to construct rules allowing to glue each element $\omega_{i,j}$ leading to the computation of $f(i)g(j)$. We suppose that $\mathcal{A}$ allows the following basic derivations:
g-computation, (C2): for all $\ell \geq 1$, $i_0, i_1, \ldots, i_\ell \geq 0$ and ω, blocking pds,

$$(q_0 b_{k+1}[T_{k,3}[a_2[{a_1}^{i_0}] b_2[{a_1}^{i_1}] \cdots b_2[{a_1}^{i_\ell}]\omega]\Omega_k] q_0 \vdash^*_{\mathcal{A}} (q_0 \mathrm{B}_{k+1}[\Omega_k] q_0)^{g(\ell)},$$

pair-generation, (G3): for all $n \geq 0$,

$$(q_0 a_{k+1}[T_{k,3}[\gamma_{n+1,n+1}\Omega_2]] q_0) \vdash^*_{\mathcal{A}} \prod_{m=0}^{n} (q_0 b_{k+1}[(T_{k,3}[\gamma_{m,n}\Omega_2])^2] q_0),$$

starting pairs, (G30):

$$(q_0 a_{k+1}[T_{k,3}[\gamma_{0,0}\Omega_2]\Omega_k] q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{B}_{k+1}[\Omega_k] q_0),$$

gluing rule, (R23): for all $0 \leq m \leq n$, there exists a blocking pds ω such that,

$$(q_0 \mathrm{B}_{k+1}[T_{k,3}[\gamma_{m,n}\Omega_2]\Omega_k] q_0) \vdash^*_{\mathcal{A}} (q_0 a_{k+1}[T_{k,3}[\gamma_{m,m}\omega\Omega_2]\Omega_k] q_0),$$

ending rule, (T4):

$$(q_0 \mathrm{B}_{k+1}[\varepsilon] q_0) \vdash^*_{\mathcal{A}} \alpha.$$

Let us prove by induction the following property $\mathbf{P}(n)$: for all $0 \leq m \leq n$ and all blocking pds ω,

$$(q_0 a_{k+1}[T_{k,3}[\gamma_{m,m}\omega]] q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{B}_{k+1}[\varepsilon] q_0)^{f(m)}.$$

Basis: $\mathbf{P}(0)$ follows from (D30), by substituting ω to Ω_2 and ε to Ω_k.
Induction step:

$$
\begin{aligned}
&(q_0 a_{k+1}[T_{k,3}[\gamma_{n+1,n+1}\omega]]q_0)\\
&\qquad (\text{by (G3)}) \vdash^*_{\mathcal{A}} \prod_{m=0}^{n} (q_0 b_{k+1}[(T_{k,3}[\gamma_{m,n}\omega])^2]q_0)\\
&\qquad (\text{by (C2)}) \vdash^*_{\mathcal{A}} \prod_{m=0}^{n} (q_0 \mathrm{B}_{k+1}[T_{k,3}[\gamma_{m,n}\omega]]q_0)^{g(n-m)}\\
&\qquad (\text{by (R23)}) \vdash^*_{\mathcal{A}} \prod_{m=0}^{n} (q_0 a_{k+1}[T_{k,3}[\gamma_{m,m}\omega_m]]q_0)^{g(n-m)}\\
&\qquad (\text{by induction hypothesis}) \vdash^*_{\mathcal{A}} \prod_{m=0}^{n} (q_0 \mathrm{B}_{k+1}[\varepsilon]q_0)^{f(m)\cdot g(n-m)}\\
&\qquad = (q_0 \mathrm{B}_{k+1}[\varepsilon]q_0)^{f(n+1)}
\end{aligned}
$$

(where all the ω_m are blocking pds). By using (T4) we obtain:

$$\forall n \in \mathbb{N}, (q_0 a_{k+1}[T_{k,2}[{a_1}^n]]q_0) \vdash^*_{\mathcal{A}} \alpha^{f(n)}.$$

Second step: We can construct an automaton $\mathcal{A}_2$ realizing (D2) and fulfilling also the condition ($\mathbf{P2}$) of Lemma 4. Starting and ending rules can be made valid by a set of transitions, in a similar way to that used in the proof of Proposition 12. The gluing rules (R23) are obtained by the following transitions:
$\Delta(q_0, \varepsilon, \mathrm{B}_{k+1} a_k \cdots a_2) = \Delta(q_0, \varepsilon, \mathrm{B}_{k+1} a_k \cdots a_2 a_1) = (\mathrm{change}_{\mathrm{B}_2}\ \mathrm{push}_{a_2}\ \mathrm{change}_{a_{k+1}}, q_0)$.
For all $\gamma_{m,n} = a_2[{a_1}^m]\omega$, we get the derivation:

$$(q_0, \mathrm{B}_{k+1}[T_{k,3}[\gamma_{m,n}]], q_0) \vdash_{\mathcal{A}} (q_0, a_{k+1}[T_{k,3}[a_2[{a_1}^m]\mathrm{B}_2[{a_1}^m]\omega]], q_0).$$

It is possible to prove this proposition for $k = 3$, but the proof is slightly different.

Proposition 15 (convolution-inverse). *Let $g \in \mathbb{S}_2$, and f the sequence defined by: $f(0) = 1$ and for all $n \geq 0$, $f(n+1) = \sum_{m=0}^{n} f(m) \cdot g(n-m)$. Then $f \in \mathbb{S}_3$.*

Proof. We use here the same notations as in the proof of the previous lemma, by adding $\gamma'_{m,n} = b_2[{a_1}^{m+1}] \cdots b_2[{a_1}^n]$ (i.e., $\gamma_{m,n} = a_2[{a_1}^m]\gamma'_{m,n}$). We recall that a blocking pushdown is a 2-pds such that $\mathrm{top}_2(\omega) = \mathrm{B}_2$. Suppose we are given an automaton $\mathcal{A} \in 3\text{-}\mathsf{CPDA}^{\boldsymbol{N}}$ for which the following derivations hold:
initialisation rule, (I0): for all $n \geq 0$,

$$(q_0, d_3[a_2[{a_1}^n]], q_0) \vdash^*_{\mathcal{A}} (q_0, a_3[\gamma_{n,n}\mathrm{B}_2[{a_1}^n]], q_0),$$

g-computation, (C2): for all $n \geq 0, m \in [1,n]$, for all blocking pds, ω

$$(q_0 b_3[\gamma'_{m,n}\omega]q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{B}_3[\omega]q_0)^{g(n-m)},$$

pair-generation, (G3): for all $n \geq 0$,

$$(q_0 a_3[a_2[{a_1}^{n+1}]\Omega_2]q_0) \vdash^*_{\mathcal{A}} \prod_{m=0}^{n} (q_0 a_3[a_2[{a_1}^m]\mathrm{B}_2[{a_1}^m]\gamma'_{m,n}\Omega_2]q_0),$$

starting pair, (G30): for all $0 \leq n$,

$$(q_0 a_3[\gamma_{0,0}\Omega_2]q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{B}_3[\Omega_2]q_0),$$

ending rule, (T0):

$$(q_0 \mathrm{A}_3[\varepsilon]q_0) \vdash^*_{\mathcal{A}} \alpha,$$

(T1):

$$(q_0 \mathrm{B}_3[\mathrm{B}_2[\Omega_1]\Omega_2]q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{A}_3[\Omega_2]q_0),$$

gluing rule, (R4): for all $\omega \neq \varepsilon \in$ 2-pds,

$$(q_0 \mathrm{A}_3[\omega]q_0) \vdash^*_{\mathcal{A}} (q_0 b_3[\omega]q_0).$$

From these rules, we prove by induction the following property $\mathbf{P}(n)$:

$$(q_0 a_3[a_2[{a_1}^n]\mathrm{B}_2[\Omega_1]\Omega_2]q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{A}_3[\Omega_2]q_0)^{f(n)}.$$

<u>*Basis*</u>: By applying (G30) then (T1), we get:

$$(q_0 a_3[\gamma_{0,0}\mathrm{B}_2[\Omega_1]\Omega_2]q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{B}_3[\mathrm{B}_2[\Omega_1]\Omega_2]q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{A}_3[\Omega_2]q_0),$$

since $\gamma_{0,0} = a_2[\varepsilon]$ and $f(0) = 1$, $\mathbf{P}(0)$ is proved.
<u>*Induction step*</u>: Suppose $\mathbf{P}(n)$, for $n \geq 0$,

$$\begin{aligned}
&(q_0 a_3[a_2[{a_1}^{n+1}]\mathrm{B}_2[\Omega_1]\Omega_2]q_0) \\
\text{(by (G3))} \vdash^*_{\mathcal{A}} &\prod_{m=0}^{n} (q_0 a_3[a_2[{a_1}^m]\mathrm{B}_2[{a_1}^m]\gamma'_{m,n}\mathrm{B}_2[\Omega_1]\Omega_2]q_0) \\
\text{(by induction hypothesis)} \vdash^*_{\mathcal{A}} &\prod_{m=0}^{n} (q_0 \mathrm{A}_3[\gamma'_{m,n}\mathrm{B}_2[\Omega_1]\Omega_2]q_0)^{f(m)} \\
\text{(by (R4))} \vdash^*_{\mathcal{A}} &\prod_{m=0}^{n} (q_0 b_3[\gamma'_{m,n}\mathrm{B}_2[\Omega_1]\Omega_2]q_0)^{f(m)} \\
\text{(by (C2))} \vdash^*_{\mathcal{A}} &\prod_{m=0}^{n} (q_0 \mathrm{B}_3[\mathrm{B}_2[\Omega_1]\Omega_2]q_0)^{f(m)g(n-m)} \\
\text{(by (T1))} \vdash^*_{\mathcal{A}} &\prod_{m=0}^{n} (q_0 \mathrm{A}_3[\Omega_2]q_0)^{f(m)g(n-m)} \\
= \ &(q_0 \mathrm{A}_3[\Omega_2]q_0)^{f(n+1)}.
\end{aligned}$$

Then, by composing the initialization rule, the derivation above and the ending rule (T0), we get for all $n \geq 0$:

$$\begin{aligned}(q_0 d_3[a_2[{a_1}^n]]q_0) &\vdash^*_{\mathcal{A}} (q_0 a_3[\gamma_{n,n}\mathrm{B}_2[{a_1}^n]]q_0)\\ &\vdash^*_{\mathcal{A}} (q_0\mathrm{A}_3[\varepsilon]q_0)\\ &\vdash^*_{\mathcal{A}} \alpha^{f(n)}.\end{aligned}$$

Second step: By proceeding as in the proof of Lemma 13, we obtain an automaton $\mathcal{A}_1 \in$ 3-DCPDA$^{\boldsymbol{N}}$ such that for all $n \geq 0$, $m \in [1, n]$

$$(q_0 b_3[\gamma'_{m,n}]q_0) \vdash^*_{\mathcal{A}} (q_0\mathrm{B}_3[\varepsilon]q_0)^{g(n-m)}$$

(it is the rule (C2) of the proof of Lemma 13) and fulfilling (**Q3**): there are no transition starting with $(q_0, \varepsilon, \mathrm{B}_3 w)$.
By adding the new symbol B_2 and by transforming the transitions in a such way as treat blocking pds as an empty pds, we obtain an automaton fulfilling (C2). This transformation does not introduce transitions starting with $(q_0, \varepsilon, \mathrm{B}_3 w)$ and (**Q3**) holds. The other derivations are obtained in an obvious way.

Remark 3. Let us see the sequence g as a formal power series

$$g = \sum_{n=0}^{\infty} g(n) X^n.$$

Proposition 14 asserts that the series $\frac{1}{1-Xg}$ belongs to $\mathbb{S}_{k+1}$. In other words, the convolution inverse of every formal power series of the form $1 - Xg$, where $g \in \mathbb{S}_k$, belongs to $\mathbb{S}_{k+1}$.

Proposition 16 (Series composition). *Let $g \in \mathbb{S}_k$, $k \geq 2$, $g(0) = 0$ and $f \in \mathbb{S}^{\boldsymbol{N}}_{k+1}$, then the sequence $(f \bullet g)(X) = f(g(X))$ belongs to $\mathbb{S}^{\boldsymbol{N}}_{k+1}$.*

Proof. For $n \geq 0$, $(f \bullet g)(n) = f(n) \cdot h(n)$ where

$$h(n) = \sum_{(i_1,\ldots,i_m)\in I_n} g(i_1)\cdots g(i_m)$$

$$\text{and } I_n = \{(i_1, \ldots, i_m) \mid m \in [1, n],\ i_1, \ldots, i_m \geq 1,\ \sum_{j\in[1,m]} i_j = n\}$$

Then, by using Proposition 7, it suffices to prove that the sequence $h(n)$ belongs to $\mathbb{S}_{k+1}$. For that, it is necessary to be able to enumerate all the m-tuples of I_n. Let us encode elements of I_n by words in $\{a_2, b_2\}^{n-1} \cdot b_2$.

All $\boldsymbol{p} = (p_1 + 1, \ldots, p_m + 1)$, $p_i \geq 0$ that belongs to I_n is associated in a bijective way to the word $u_{\boldsymbol{p}} = {a_2}^{p_1} b_2 {a_2}^{p_2} b_2 \cdots {a_2}^{p_m} b_2$ in $\{a_2, b_2\}^{n-1} \cdot b_2$.

We can then represent every element of I_n by a 2-pds by using the following encoding of the words belonging to $\{a_2, b_2\}^*$:

$\forall u = \beta_1 \cdots \beta_n \in \{a_2, b_2\}^n$,

$$\gamma_u = \beta_1[a_1]\beta_2[a_1 a_1]\cdots\beta_n[{a_1}^n]c_2[{a_1}^n].$$

Now, we need an order to enumerate all the m-tuples of I_n. We choose the reversed lexical order: for all $u, v \in \{a_2, b_2\}^n$, we write $u <_n v$ iff $u = u' a_2 w$ and $v = v' b_2 w$. We get then a total order over elements $\{a_2, b_2\}^{n-1} \cdot b_2$. The least element is ${a_2}^{n-1} b_2$ and the greatest element is ${b_2}^n$. We define the successor relation for this order: if $u = {b_2}^p a_2 w$, the successor of u is $succ_{lex}(u) = {a_2}^p b_2 w$.

First step: We suppose we are given $\mathcal{A} = (Q, \{\alpha\}, \mathcal{A}_{k+1}, \Delta, q_0, a_{k+1}) \in (k+1)$-DCPDA, with $A_{k+1} \supseteq \{a_{k+1}, \mathrm{A}_{k+1}, b_{k+1}, c_{k+1}\}$, $A_i \supseteq \{a_i\}$ for $i \in [k, 3]$, $A_2 \supseteq \{a_2, b_2, c_2\}$ and $A_1 = \{a_1\}$, allowing the following basic derivations:
initialization, (I0): $\forall n \geq 0$,

$$(q_0 a_{k+1}[T_{k,3}[c_2[{a_1}^n]]]q_0) \vdash^*_{\mathcal{A}} (q_0 c_{k+1}[T_{k,3}[\gamma_{{a_2}^{n-1} b_2}]]]q_0),$$

m-tuple-generation, (G1): for all $u \in \{a_2, b_2\}^{n-1} \cdot b_2$, $u \neq {b_2}^n$,

$$(q_0 c_{k+1}[T_{k,3}[\gamma_u]]q_0) \vdash^*_{\mathcal{A}} (q_0 c_{k+1}[T_{k,3}[\gamma_{succ_n(u)}]]q_0)(q_0 b_{k+1}[T_{k,3}[\gamma_u]]q_0),$$

m-tuple-generation, (G10):

$$(q_0 c_{k+1}[T_{k,3}[\gamma_{{b_2}^n}]]q_0) \vdash^*_{\mathcal{A}} (q_0 b_{k+1}[T_{k,3}[\gamma_{{b_2}^n}]]q_0),$$

g-computation, (D2): for all $\ell \geq 1$, $i_0, i_1, \ldots, i_\ell \geq 0$,

$$(q_0 b_{k+1}[T_{k,3}[a_2[{a_1}^{i_0}]a_2[{a_1}^{i_1}]\cdots a_2[{a_1}^{i_{\ell-1}}]b_2[{a_1}^{i_\ell}]\Omega_2]]q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{A}_{k+1}[T_{k,3}[\Omega_2]]q_0)^{g(\ell+1)},$$

g-computation, (D20): for all $i \geq 0$,

$$(q_0 b_{k+1}[T_{k,3}[b_2[{a_1}^i]\Omega_2]]q_0) \vdash^*_{\mathcal{A}} (q_0 \mathrm{A}_{k+1}[T_{k,3}[\Omega_2]]q_0)^{g(1)},$$

ending rules, (T0): for all $i \geq 0$,

$$(q_0 \mathrm{A}_{k+1}[T_{k,3}[c_2[{a_1}^i]]]q_0) \vdash^*_{\mathcal{A}} \alpha$$

ending rules, (T1): for all $i \geq 0$,

$$(q_0 \mathrm{A}_{k+1}[T_{k,3}[a_2[\Omega_1]\Omega_2]]q_0) \vdash^*_{\mathcal{A}} (q_0 b_{k+1}[T_{k,3}[a_2[\Omega_1]\Omega_2]]q_0),$$

ending rules, (T1'): for all $i \geq 0$

$$(q_0 \mathrm{A}_{k+1}[T_{k,3}[b_2[\Omega_1]\Omega_2]]q_0) \vdash^*_{\mathcal{A}} (q_0 b_{k+1}[T_{k,3}[b_2[\Omega_1]\Omega_2]]q_0).$$

Let us prove that such an automaton computes the sequence h. Since $<_n$ defines a total order over $\{a_2, b_2\}^{n-1} b_2$ whose the least element is ${a_2}^{n-1} b_2$ and the greatest is ${b_2}^n$, by applying the initialization rule (I0), then iteration of (G1), then (G10), we get:

$$\begin{aligned}(q_0 a_{k+1}[T_{k,3}[c_2[{a_1}^n]]]q_0) &\vdash^*_{\mathcal{A}} (q_0 c_{k+1}[T_{k,3}[\gamma_{{a_2}^{n-1} b_2}]]]q_0)\\ &\vdash^*_{\mathcal{A}} \prod_{\boldsymbol{p} \in I_n} (q_0 b_{k+1}[T_{k,3}[\gamma_{u_{\boldsymbol{p}}}]]q_0).\end{aligned} \tag{14}$$

Starting from each factor of the product (14) encoding $\boldsymbol{p} = (p_1, \ldots, p_m)$, $p_i \geq 1$, we get, by applying iteratively (C2,T1) and (C20,T1) (or (C2) when $\gamma_{u_{p_i,\ldots,p_m}}$ start with a_2 and (C20) when $\gamma_{u_{p_i,\ldots,p_m}}$ start with b_2):

$$\begin{aligned}
(q_0 b_{k+1}[T_{k,3}[\gamma_{u_{\boldsymbol{p}}}]]q_0) &\vdash^*_{\mathcal{A}} (q_0 \mathrm{A}_{k+1}[T_{k,3}[\gamma_{u_{p_2,\ldots,p_m}}]]q_0)^{g(p_1)} \\
&\vdash^*_{\mathcal{A}} (q_0 b_{k+1}[T_{k,3}[\gamma_{u_{p_2,\ldots,p_m}}]]q_0)^{g(p_1)} \\
&\vdash^*_{\mathcal{A}} \cdots \\
&\vdash^*_{\mathcal{A}} (q_0 \mathrm{A}_{k+1}[T_{k,3}[c_2[{a_1}^n]]]q_0)^{g(p_1)\cdots g(p_m)}. \qquad (15)
\end{aligned}$$

Let us apply to each element of the product (14), the associated derivation (15). We get

$$\begin{aligned}
(q_0 a_{k+1}[T_{k,3}[c_2[{a_1}^n]]]q_0) &\vdash^*_{\mathcal{A}} \prod_{(p_1,\ldots,p_m)\in I_n} (q_0 \mathrm{A}_{k+1}[T_{k,3}[c_2[{a_1}^n]]]q_0)^{g(p_1)\cdots g(p_m)} \\
&\vdash^*_{\mathcal{A}} \prod_{(p_1,\ldots,p_m)\in I_n} \alpha^{g(p_1)\cdots g(p_m)} \text{ (by rule (T0))} \\
&= \alpha^{\sum_{(p_1,\ldots,p_m)\in I_n} g(p_1)\cdots g(p_m)} = \alpha^{h(n)}.
\end{aligned}$$

The automaton $\mathcal{A}$ computes then the sequence h.

Second step: Let us construct a such an automaton. Using a construction similar to that given in the proof of Proposition 14 (and similar to that given in the proof of Proposition 15 if $k = 2$) for the derivation (C2), we get $\mathcal{A}_2 = (Q_2, \emptyset, (B_1, \ldots, B_{k+1}), \Delta_2, q_0, b_{k+1}) \in (k+1)$-DCPDA, where $B_{k+1} \supseteq \{b_{k+1}, \mathrm{A}_{k+1}\}$, $B_i \supseteq \{a_i\}$ for $i \in [3, k]$, $B_2 \supseteq \{a_2, b_2, c_2\}$ and $B_1 = \{a_1\}$ and fulfilling derivations (C2) and (C20) and conditions (**Q2**) and ($\mathbf{Q2}(\mathrm{A}_{k+1})$) stated in Lemma 4.

We define $\mathcal{A} = (Q, \{\alpha\}, (A_1, \ldots, A_{k+1}), \Delta, q_0, a_{k+1})$ where

$$Q = Q_2 \cup \{q_1, q_2\}, \quad A_i = B_i \text{ for } i \in [1, k], \quad A_{k+1} = B_{k+1} \cup \{c_{k+1}, a_{k+1}\},$$

and Δ is the union of Δ_2 with the following transitions: for $w = a_k \cdots a_3$,

(0.1) $\Delta(q_0, \varepsilon, a_{k+1} w c_2 a_1) = (\mathrm{push}_{b_2}, q_1)$,
(0.2) $\Delta(q_1, \varepsilon, a_{k+1} w a_2 a_1) = \Delta(q_1, \varepsilon, a_{k+1} w b_2 a_1) = (\mathrm{push}_{a_2}\, \mathrm{pop}_1, q_1)$,
(0.3) $\Delta(q_1, \varepsilon, a_{k+1} w a_2) = \Delta(q_1, \varepsilon, a_{k+1} w b_2) = (\mathrm{pop}_2\, \mathrm{change}_{c_{k+1}}, q_0)$,

(1.1.1) $\Delta(q_0, \varepsilon, c_{k+1} w b_2 a_1) = (\mathrm{change}_{b_{k+1}}\, \mathrm{push}_{c_{k+1}}, q_2)$,
(1.1.2) $\Delta(q_0, \varepsilon, c_{k+1} w a_2 a_1) = (\mathrm{change}_{b_{k+1}}\, \mathrm{push}_{c_{k+1}}\, \mathrm{change}_{b_2}, q_0)$,
(1.2.1) $\Delta(q_2, \varepsilon, c_{k+1} w b_2 a_1) = (\mathrm{pop}_2, q_2)$,
(1.3.0) $\Delta(q_2, \varepsilon, c_{k+1} w c_2 a_1) = (\mathrm{pop}_{k+1}, q_0)$,
(1.3.1) $\Delta(q_2, \varepsilon, c_{k+1} w a_2 a_1) = (\mathrm{change}_{b_2}, q_1)$,
(1.4.1) $\Delta(q_1, \varepsilon, c_{k+1} w b_2 a_1) = \Delta(q_1, \varepsilon, c_{k+1} w a_2 a_1) = (\mathrm{pop}_1\, \mathrm{push}_{a_2}, q_1)$,
(1.5.1) $\Delta(q_1, \varepsilon, c_{k+1} w a_2) = (\mathrm{pop}_2, q_0)$,

(3) $\Delta(q_0, \alpha, \mathrm{A}_{k+1} w c_2 a_1) = (\mathrm{pop}_{k+1}, q_0)$,

(4) $\Delta(q_0, \alpha, \mathrm{A}_{k+1} w b_2 a_1) = \Delta(q_0, \alpha, \mathrm{A}_{k+1} w a_2 a_1) = (\mathrm{change}_{b_{k+1}}, q_0)$.

These transitions are all incompatibles between them. Since c_{k+1} and a_{k+1} are new symbols, and $\mathcal{A}_2$ is deterministic and fulfills the condition ($\mathbf{Q3}(\mathrm{A}_{k+1})$), the addition of transitions Δ_1 does not modify the determinism of Δ. The automaton $\mathcal{A}$ is then deterministic.

Now, we prove that $\mathcal{A}$ realize the derivations describe in the first step.

Derivations (C2) and (C20) hold by transitions Δ_2. The ending rule (T0) is obtained by applying the transition (3), derivations (T1) and (T1') hold by choice of transitions (4).

By transitions (0.1), then iteration of (0.2), then (0.3) we get the following derivation:

$$\begin{aligned}(q_0 a_{k+1}[T_{k,3}[c_2[{a_1}^n]]]q_0) &\vdash_{\mathcal{A}} (q_1 a_{k+1}[T_{k,3}[b_2[{a_1}^n]c_2[{a_1}^n]]]q_0)\\ &\vdash^*_{\mathcal{A}} (q_1, a_{k+1}[T_{k,3}[a_2[\varepsilon]a_2[a_1]\cdots a_2[{a_1}^{n-1}]b_2[{a_1}^n]c_2[{a_1}^n]]]q_0)\\ &\vdash_{\mathcal{A}} (q_0, c_{k+1}[T_{k,3}[a_2[a_1]\cdots a_2[{a_1}^{n-1}]b_2[{a_1}^n]c_2[{a_1}^n]]]q_0)\\ &= (q_0 c_{k+1}[T_{k,3}[\gamma_{{a_2}^{n-1}b_2}]]q_0).\end{aligned}$$

Then the initial derivation (I0) holds.

Let us prove that generation rules hold. To verify (G1), we distinguish two cases according to the first letter of $u \in \{a_2, b_2\}^{n-1} \cdot b_2$, $u \neq {b_2}^n$:

Case 1: if $u = {b_2}^i a_2 w$, then, let ω be the 2-pds such that $\gamma_u = b_2[a_1]\cdots b_2[{a_1}^i]a_2[{a_1}^{i+1}]\omega$, we get the following derivation:

$$\begin{aligned}& (q_0 c_{k+1}[T_{k,3}[\gamma_u]]q_0)\\ \text{(by 1.1.1)} &\vdash_{\mathcal{A}} (q_2 c_{k+1}[T_{k,3}[\gamma_u]]q_0)(q_0 b_{k+1}[T_{k,3}[\gamma_u]]q_0)\\ \text{(by (1.2.1))} &\vdash^i_{\mathcal{A}} (q_2 c_{k+1}[T_{k,3}[a_2[{a_1}^{i+1}]\omega]]q_0)(q_0 b_{k+1}[T_{k,3}[\gamma_u]]q_0)\\ \text{(by (1.3.1))} &\vdash_{\mathcal{A}} (q_1 c_{k+1}[T_{k,3}[b_2[{a_1}^{i+1}]\omega]]q_0)(q_0 b_{k+1}[T_{k,3}[\gamma_u]]q_0)\\ \text{(by (1.4.1))} &\vdash^i_{\mathcal{A}} (q_1 c_{k+1}[T_{k,3}[a_2[\varepsilon]\cdots a_2[{a_1}^i]b_2[{a_1}^{i+1}]\omega]]q_0)(q_0, b_{k+1}[T_{k,3}[\gamma_u]]q_0)\\ \text{(by (1.5.1))} &\vdash_{\mathcal{A}} (q_0 c_{k+1}[T_{k,3}[\gamma_{succ_{lex}(u)}]]q_0)(q_0 b_{k+1}[T_{k,3}[\gamma_u]]q_0).\end{aligned}$$

Case 2: if $u = a_2 w$, then, let ω such that $\gamma_u = a_2[a_1]\omega$.

We get the following derivation:

$$\begin{aligned}& (q_0 c_{k+1}[T_{k,3}[\gamma_u]]q_0)\\ \text{(by (1.1.2))} &\vdash_{\mathcal{A}} (q_0 c_{k+1}[T_{k,3}[b_2[a_1]\omega]]q_0)(q_0 b_{k+1}[T_{k,3}[\gamma_u]]q_0)\\ &= (q_0 c_{k+1}[T_{k,3}[\gamma_{succ_{lex}(u)}]]b_{k+1}[T_{k,3}[\gamma_u]]q_0).\end{aligned}$$

Finally, let us prove that the generation rule (G10) holds:

$$\begin{aligned}& (q_0 c_{k+1}[T_{k,3}[\gamma_{{b_2}^n}]]q_0)\\ \text{(by (1.1.1))} &\vdash_{\mathcal{A}} (q_2 c_{k+1}[T_{k,3}[\gamma_{{b_2}^n}]]q_0)(q_0 b_{k+1}[T_{k,3}[\gamma_{{b_2}^n}]]q_0)\\ \text{(by (1.2.1))} &\vdash^n_{\mathcal{A}} (q_2 c_{k+1}[T_{k,3}[c_2[{a_1}^n]]]q_0)(q_0 b_{k+1}[T_{k,3}[\gamma_{{b_2}^n}]]q_0)\\ \text{(by (1.3.0))} &\vdash_{\mathcal{A}} (q_0 b_{k+1}[T_{k,3}[\gamma_{{b_2}^n}]]q_0).\end{aligned}$$

Proposition 17 (Sequence composition). *Let $k_1, k_2 \geq 1$, $f \in \mathbb{S}_{k_1+1}$ and $g \in \mathbb{S}^{\boldsymbol{N}}_{k_2+1}$, then $g \circ f \in \mathbb{S}^{\boldsymbol{N}}_{k_1+k_2+1}$.*

Construction: Using Lemma 3, after a suitable choice for the concrete sets of states and pushdown alphabets we obtain $\mathcal{A}_1 \in (k_1+1)\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}(A_1, \ldots, A_{k_1+1})$, and $\mathcal{A}_2 \in (k_2+1)\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}(B_1, \ldots, B_{k_2+1})$ fulfilling conditions:
(**P1.1**) $\forall \Omega_k \in \mathcal{I}_k, (q_0, a_{k_1+1}[a_{k_1}[\cdots[a_2[a_1{}^n]]\cdots]], q_0) \vdash^*_{\mathcal{A}_1} (q_0, \mathrm{A}_{k_1+1}[\varepsilon], q_0)^{f(n)}$.
(**P1.2**) $\forall \Omega_k \in \mathcal{I}_k, (r_0, b_{k_2+1}[b_{k_2}[\cdots[b_2[b_1{}^n]]\cdots]], r_0) \vdash^*_{\mathcal{A}_2} (r_0, \mathrm{B}_{k_2+1}[\varepsilon], r_0)^{g(n)}$.
(**P2.1**) Δ_1 does not contain lefthand side of the form $(q_0, \varepsilon, \varepsilon)$, $q \in Q_1$.
(**P2.2**) Δ_2 does not contain lefthand side of the form $(q, \varepsilon, \varepsilon)$, $q \in Q_2$.
(**P3.1**) Δ_1 does not contain lefthand side of the form $(q, \varepsilon, \mathrm{A}_{k+1} \cdot w)$.
(**P3.2**) Δ_2 does not contain lefthand side of the form $(r_0, \varepsilon, \mathrm{B}_{k+1} \cdot w, \boldsymbol{o})$.
(**P4**) $A_1 \cap B_{k_2+1} = \{a_1\} = \{\mathrm{B}_{k_2+1}\}$.
We consider $\mathcal{A} = (Q, \{\alpha\}, (C_1, \ldots, C_{k_1+k_2+1}), \boldsymbol{N}, \Delta, (q_0, r_0), a_{k_1+1}) \in (k_1+k_2+1)\text{-}\mathsf{DCPDA}^{\boldsymbol{N}}$ where: $Q = Q_1 \times Q_2$ and $C_{k_2+k_1+1} = A_{k_1+1}, \ldots, C_{k_2+2} = A_2, C_{k_2+1} = B_{k_2+1}, \ldots, C_1 = B_1$ and Δ is the union of the following transitions:

Transitions inherited from $\mathcal{A}_1$:

for all $\Delta_1(q_1, \varepsilon, w_1) = (\text{instr}, p_1)$, $w_1 \in \text{top}((k_1+1)\text{-}\mathsf{pds})$, for all $\boldsymbol{o} \in \{0,1\}^{|\boldsymbol{N}|}$,
(1) $\Delta((q_1, r_0), \varepsilon, w_1, \boldsymbol{o}) = (\text{instr}^{+k_2}, (p_1, r_0))$,
where the notation instr^{+k} means:

- if instr $= \text{pop}_i$ then $\text{instr}^{+k} = \text{pop}_{i+k}$
- else $\text{instr}^{+k} = \text{instr}$.

Transitions inherited from $\mathcal{A}_2$:

for all $\Delta_2(r, \varepsilon, w_2, \boldsymbol{o}) = (\text{instr}, r')$, $w_2 \in \text{top}((k_2+1)\text{-}\mathsf{pds})$, $r, r' \in Q_2$, and for all $q_1 \in Q_1$, $w_1 \in A_{k_1+1} \cdots A_2$:
(2) $\Delta((q_1, r), \varepsilon, w_1 \cdot w_2, \boldsymbol{o}) = (\text{instr}, (q_1, r'))$.

Proof. Let us prove that the above automaton $\mathcal{A}$ is deterministic.
The fact that the initial automata $\mathcal{A}_i$ $(i = 1, 2)$ are deterministic, entails that no pair of transitions of the same type are incompatible. Now, suppose that there is one transition of type (1) compatible with a transition of type (2). We would then have

$$((q, r_0), w_1) = ((q_1, r), w'_1 \cdot w'_2),$$

for $q, q_1 \in Q_1$, $r \in Q_2$, $w_1 \in \text{top}((k_1+1)\text{-}\mathsf{pds}(A_1, \ldots, A_{k_1+1}))$, $w'_1 \in A_{k_1+1} \cdots A_1$, $w'_2 \in \text{top}((k_2+1)\text{-}\mathsf{pds})(B_1, \ldots, B_{k_2+1})$. The only possibility for a such an equality is that

$$r = r_0 \text{ and } [w'_2 = a_1 = \mathrm{B}_{k_2+1} \text{ or } w'_2 = \varepsilon]$$

But, by condition (**P3.2**), there is no transition of Δ_2 starting with $(r_0, \mathrm{B}_{k_2+1})$, in the same way, by (**P2.2′**), there is no transition of Δ_2 starting with (r_0, ε). Finally, we are sure that $\mathcal{A}$ is deterministic.
Let us check now that

$$((q_0, r_0) a_{k_1+1}[\cdots a_2[b_{k_2+1}[\cdots[b_2[b_1{}^n]]\cdots]]\cdots](q_0, r_0)) \vdash^*_{\mathcal{A}} ((q_0, r_0)\mathrm{A}_{k_1+1}[\varepsilon](q_0, r_0))^{f(g(n))} \quad (16)$$

In order to show such a derivation, we introduce the partial map $\varphi : (k_2+1)\text{-}\mathsf{pds}(B_1, \ldots, B_{k_2+1}) \to 1\text{-}\mathsf{pds}(A_1)$ fulfilling for all $\omega \in (k_2+1)\text{-}\mathsf{pds}(B_1, \ldots, B_{k_2+1})$:

$$\varphi(\omega) = {a_1}^n \Leftrightarrow (r_0 \omega r_0) \vdash^*_{\mathcal{A}_2} (r_0 \mathrm{B}_{k_2+1} r_0)^n \text{ or } \omega = \varepsilon \text{ and } \varphi(\omega) = \varepsilon. \tag{17}$$

Hence, $\varphi(\omega)$ is defined exactly for those ω such that $(r_0 \omega r_0)$ derived (modulo $\mathcal{A}_2$) into $(r_0 \mathrm{B}_{k_2+1} r_0)^*$.

Lemma 5. *For all (k_1+1)-term $T_1[\Omega_1]$, where Ω_1 is an indeterminate of level 1 admitting one and only one occurrence in T_1, for all $\omega_1 \in \mathrm{dom}(\varphi)$ and $\omega \in (k_1+1)\text{-}\mathsf{pds}$,*

$$\text{if } (q, \varepsilon, T_1[\varphi(\omega_1)]) \to_{A_1} (q', \varepsilon, \omega),$$

then, there exists $\omega_2 \in \mathrm{dom}(\varphi)$ and a (k_1+1)-term $T_2[\Omega_1]$, such that for all $p \in Q$

$$((q, r_0){T_1}^{+k_2}[\omega_1](p, r_0)) \vdash^*_{\mathcal{A}} ((q', r_0){T_2}^{+k_2}[\omega_2](p, r_0)) \text{ and } T_2[\varphi(\omega_2)] = \omega.$$

Proof. Let us suppose that the applied transition is $(q, \varepsilon, w, \mathrm{instr}, q')$, i.e.,

$$(q, \varepsilon, T_1[\varphi(\omega_1)]) \to_{A_1} (q', \varepsilon, \mathrm{instr}(T_1[\varphi(\omega_1)])) \text{ and } \omega = \mathrm{instr}(T_1[\varphi(\omega_1)]). \tag{18}$$

Let us distinguish two cases, depending on the position of the indeterminate in T_1:

Case 1: if Ω_1 does not appear in top symbols of T_1, then $\mathrm{instr}(T_1)$ is defined. Let $T_2[\Omega_1] = \mathrm{instr}(T_1[\Omega_1])$. We have then $\omega = T_2[\varphi(\omega_1)]$.

Since $\mathrm{top}_1(T) \neq \Omega_1$, $|\,\mathrm{top}({T_1}^{+k_2}[\omega])| \leq k_1 + 1$, and by using the transition of type (1) associated to the transition applied in (18) (and having for test $\chi_{\boldsymbol{N}}(0)$), we get

$$((q, r_0){T_1}^{+k_2}[\omega_1](p, r_0)) \vdash_{\mathcal{A}} ((q', r_0){T_2}^{+k_2}[\omega_1](p, r_0)).$$

Case 2: if the occurrence of Ω_1 appears in top symbols of T_1, we distinguish two sub-cases depending on the level of the instruction instr.

Case 2.1 if instr is an instruction of level greater than 1, then $\mathrm{instr}(T_1)$ is defined. Let $T_2 = \mathrm{instr}(T_1)$, then $\mathrm{instr}^{+k_2}({T_1}^{+k_2}) = {T_2}^{+k_2}$ and $\omega = T_2[\varphi(\omega_1)]$. We distinguish again two subcases depending on the value of $\varphi(\omega_1)$:

Case 2.1.1: if $\varphi(\omega_1) = \varepsilon$, then, by applying the transitions of type (2), then the transition of type (1) associated to the computation (18):

$$\begin{aligned} ((q, r_0){T_1}^{+k_2}[\omega_1](p, r_0)) &\vdash^*_{\mathcal{A}} ((q, r_0){T_1}^{+k_2}[\varepsilon](p, r_0)) \\ &\vdash^*_{\mathcal{A}} ((q', r_0)\,\mathrm{instr}^{+k_2}({T_1}^{+k_2})[\varepsilon](p, r_0)). \end{aligned}$$

The lemma holds then for $\omega_2 = \varepsilon$ since $T_2[\varphi(\omega_2)] = T_2[\varepsilon] = \omega$.

Case 2.1.2: if $\varphi(\omega_1) = {a_1}^{n+1}$, $n \geq 0$, then by definition of φ, there exists $\hat{\omega}_1 \in \mathrm{dom}(\varphi)$ such that:

$$(r_0 \omega_1 r_0) \vdash^*_{\mathcal{A}_2} (r_0 \mathrm{B}_{k_2+1} r_0)(r_0 \hat{\omega}_1 r_0) \text{ et } \varphi(\hat{\omega}_1) = {a_1}^n.$$

By applying the transitions of type (2), then the transition of type (1) associated to the computation (18):

$$(q, r_0)T_1{}^{+k_2}[\omega_1](p, r_0)) \vdash^*_{\mathcal{A}_1} ((q, r_0)T_1{}^{+k_2}[\mathrm{B}_{k_2+1}\hat{\omega}_1](p, r_0))$$
$$\vdash^*_{\mathcal{A}_1} ((q', r_0)\,\mathrm{instr}^{+k_2}(T_1{}^{+k_2}[\mathrm{B}_{k_2+1}\hat{\omega}_1])(p, r_0)).$$

Then the lemma holds for $\omega_2 = \mathrm{B}_{k_2+1}\hat{\omega}_1$ since $T_2[\varphi(\omega_2)] = T_2[a_1{}^{n+1}] = \omega$.

Case 2.2 if instr is an instruction of level 1, let $T_2 = T_1$. We get then $\omega = T_2[\mathrm{instr}(\varphi(\omega_1))]$. We distinguish again two cases depending on the value of $\varphi(\omega_1)$:

Case 2.2.1: if $\varphi(\omega_1) = \varepsilon$, then by applying transitions of type (2), then the transition of type (1) associated to the computation (18):

$$(q, r_0)T_1{}^{+k_2}[\omega_1](p, r_0)) \vdash^*_{\mathcal{A}_1} ((q, r_0)T_1{}^{+k_2}[\varepsilon](p, r_0))$$
$$\vdash^*_{\mathcal{A}_1} ((q', r_0)T_1{}^{+k_2}[\mathrm{instr}^{+k_2}(\varepsilon)](p, r_0))$$

(in this case instr = push_{a_1}). Then, the lemma holds for $\omega_2 = \mathrm{instr}^{+k_2}(\varepsilon)$ since $T_2[\varphi(\omega_2)] = T_2[\varphi(\varepsilon)] = \omega$.

Case 2.2.2: if $\varphi(\omega) = a_1{}^{n+1}$, $n \geq 0$, then by applying the transitions of type (2), then the transition of type (1) associated to the computation (18):

$$(q, r_0)T_1{}^{+k_2}[\omega_1](p, r_0)) \vdash^*_{\mathcal{A}_1} ((q, r_0)T_1{}^{+k_2}[\mathrm{B}_{k_2+1}\hat{\omega}_1](p, r_0))$$
$$\vdash^*_{\mathcal{A}_1} ((q', r_0)T_1{}^{+k_2}[\mathrm{instr}^{+k_2}(\mathrm{B}_{k_2+1}\hat{\omega}_1)](p, r_0)).$$

Then the lemma holds for $\omega_2 = \mathrm{instr}^{+k_2}(\mathrm{B}_{k_2+1}\hat{\omega}_1)$ since

$$T_2[\varphi(\omega_2)] = T_2[\mathrm{instr}(a_1{}^{n+1})] = \omega.$$

We extend now this lemma to derivations by introducing a partial function $\Phi : V_{\mathcal{A}} \to V_{\mathcal{A}_1}$, from the set of variables of $\mathcal{A}$ (defined in §4.1 by equation (1)) to the set of variables $\mathcal{A}_1$. For all $T[\Omega_1, \ldots, \Omega_n] \in \mathcal{T}_{(k_1+1)}(A_1, \ldots, A_{k_1+1})$, Ω_i indeterminates of level 1 and $\omega_1, \ldots, \omega_n \in (k_2+1)\text{-}\mathsf{pds}(B_1, \ldots, B_{k_2+1})$, $q \in Q_1$, we define

$$\Phi((q, r_0)T^{+k_2}[\omega_1, \ldots, \omega_n](q', r_0)) = (qT[\varphi(\omega_1), \ldots, \varphi(\omega_n)]q')$$

hence $\Phi(V)$ is defined exactly for variables $V = ((q, r_0)T^{+k_2}[\omega_1, \ldots, \omega_n](q', r_0))$ such that for every Ω_i appearing in T, $\omega_i \in \mathrm{dom}(\varphi)$. We extend the map Φ to words by setting:

$$\Phi(V_1V_2\cdots V_m) = \Phi(V_1)\Phi(V_2)\cdots\Phi(V_m) \text{ if for all } i\ , V_i \in \mathrm{dom}(\Phi)$$

and $\Phi(V_1V_2\cdots V_m)$ is undefined otherwise.

Lemma 6. *If* $U \in \mathrm{dom}(\Phi)$ *and* $U_1' \in V_{\mathcal{A}_1}{}^*$ *are such that*

$$\Phi(U) \vdash_{\mathcal{A}_1} U_1'$$

then, there exists a word $U' \in \mathrm{dom}(\Phi)$ *such that*

$$U \vdash^*_{\mathcal{A}} U' \ \ \& \ \ \Phi(U') = U_1'.$$

Proof. Let us prove this lemma. It is sufficient to prove it in the case where U is reduced to one variable. Suppose $U = (q, r_0)T[\omega_1, \omega_2, \ldots, \omega_n](q', r_0)$ where $T[\Omega_1, \Omega_2, \ldots, \Omega_n]$ is a $(k_1 + 1)$-term, each ω_i belongs to $\mathrm{dom}(\varphi)$ and $q, q' \in Q_1$. Without loss of generality, we can suppose that each Ω_i has exactly one occurrence in T and for all $1 \leq i < j \leq n$, the occurence of Ω_i is on the left of the occurence of Ω_j. We suppose that

$$\Phi(U) \vdash_{\mathcal{A}_1} U_1'. \tag{19}$$

Let us distinguish three cases, depending on the type of rule used in derivation (19).
<u>*Case 1*</u>: decomposition rule.
This means that $T = T' \cdot T''$ and then, there exists $n' \in [1, n]$ such that

$$\begin{aligned} qT[\varphi(\omega_1), \ldots, \varphi(\omega_n)]q' \vdash_{\mathcal{A}_1} & \; qT_1[\varphi(\omega_1), \ldots, \varphi(\omega_{n'})]q'' \cdot q''T_2[\varphi(\omega_{n'+1}), \ldots, \varphi(\omega_n)]q' \\ = & \; U_1'. \end{aligned}$$

In this case, by decomposition rule, $U \vdash^*_{\mathcal{A}} U'$ with

$$U' = (q, r_0){T_1}^{+k_2}[\omega_1, \ldots, \omega_{n'}](q'', r_0) \cdot (q'', r_0){T_2}^{+k_2}[\omega_{n'+1}, \ldots, \omega_n](q', r_0)$$

fulfills the conclusion of the lemma.
<u>*Case 2*</u>: transition rule.
There exists $p \in Q_1$ and an instruction instr such that (19) can be translated in:

$$(q, T[\varphi(\omega_1), \ldots, \varphi(\omega_n)], q') \vdash_{\mathcal{A}_1} (p, \mathrm{instr}(T[\varphi(\omega_1), \ldots, \varphi(\omega_n)]), q').$$

By applying the lemma above, there exists $U' \in \mathrm{dom}(\Phi)$ such that $U \vdash^*_{\mathcal{A}_1} U'$ and $\Phi(U') = (p, T[\varphi(\omega_1), \ldots, \varphi(\omega_n)], q')$.

We can now prove the derivation (16). Let $k = k_1 + k_2 + 1$, we remark that,

$$\begin{aligned} \Phi((q_0, r_0){T_{k,k_2+1}}^{+k_2}[b_{k_2}[\cdots b_2[{b_1}^n] \cdots]](q_0, r_0)) &= (q_0 T_{k,k_2+1}[{a_1}^{g(n)}]q_0) \\ &\vdash^*_{\mathcal{A}_1} (q_0 \mathrm{A}_{k_1+1}[\varepsilon]q_0)^{f(g(n))}. \end{aligned}$$

Applying Lemma 6 iteratively, we obtain some $U' \in \mathrm{dom}(\Phi)$ such that:

$$((q_0, r_0){T_{k,k_2+1}}^{+k_2}[b_{k_2}[\cdots b_2[{b_1}^n] \cdots]](q_0, r_0)) \vdash^*_{\mathcal{A}} U' \quad \text{and} \quad \Phi(U') = (q_0 \mathrm{A}_{k_1+1}[\varepsilon]q_0)^{f(g(n))}.$$

But the only possible value for a pre-image by Φ of $(q_0 \mathrm{A}_{k_1+1}[\varepsilon]q_0)^{f(g(n))}$ is

$$U' = ((q_0, r_0)\mathrm{A}_{k_1+1}[\varepsilon](q_0, r_0))^{f(g(n))},$$

since $\mathrm{A}_{k+1}[\varepsilon]$ cannot correspond to term without indeterminate of level 1.

4.4 Proofs of the Section 3

Theorem 7:

Proof. It is possible to construct a k-$\mathsf{DCPDA}^{\boldsymbol{N}}$ recognizing the language $L \in (\{\alpha\} \cup \{\beta_{\boldsymbol{o}} \mid \boldsymbol{o} \in \{0,1\}^m\})^*$:

$$L = \{\alpha^{s(0)} x_0 \cdots \alpha^{s(n)} x_n \mid n \geq 0,\, \forall i \in [1,n],\, x_i = \beta_{\chi_{\boldsymbol{N}}(i)}\}$$

and whose computation graph consists of an infinite path labelled by the word

$$\alpha^{s(0)} \beta_{\chi_{\boldsymbol{N}}(0)} \cdots \alpha^{s(n)} \beta_{\chi_{\boldsymbol{N}}(n)} \cdots$$

Let $P_{\boldsymbol{o}} = \{n \mid \chi_{\boldsymbol{N}}(n) = \boldsymbol{o}\}$. The structure $\mathcal{S} = \langle \mathbb{N},\, +1,\, \Sigma s(\mathbb{N}),\, (\Sigma s(P_{\boldsymbol{o}}))_{\boldsymbol{o} \in \{0,1\}^m} \rangle$ is interpretable in this graph. From Theorem 6, since $\langle \mathbb{N}, +1, \boldsymbol{N} \rangle$ has a decidable theory, the structure $\mathcal{S}$ so has.

Finally, $\langle \mathbb{N}, +1, \Sigma s(\mathbb{N}),\, \Sigma s(N_1), \ldots,\, \Sigma s(N_m) \rangle$ is clearly interpretable in $\mathcal{S}$ since for every $i \in [1, m]$,

$$\Sigma s(N_i) = \bigcup_{\boldsymbol{o} \mid \pi_i(\boldsymbol{o}) = 1} \Sigma s(P_{\boldsymbol{o}})$$

and has then a decidable MSO-theory.

Proposition 4:

Proof. Suppose that $P(X) = \sum_{i=0}^{m} a_i X^i$. From definition, $u(n) = P(n) \in \Sigma \mathbb{S}_2$ iff there exists $v \in \mathbb{S}_2$ such that $\Sigma v = u$.

Let v be the sequence defined by: $v(0) = u(0)$ and for all $n \geq 0$, $v(n+1) = \sum_{i=0}^{m} a_i \sum_{j=0}^{i-1} \binom{j}{i} n^j$.[2]

Clearly, $v \in \mathbb{S}_2$ since it is $\mathbb{N}$-rational (Proposition 1) and $\Sigma v = u$.

Proposition 5:

Proof. Let s be a strictly increasing sequence. We consider the sequence u defined by: $u(0) = 0$ and for all $n \geq 0$, $u(n+1) = 1$ if $n+1 \in s(\mathbb{N}^+)$ and $u(n+1) = 0$ if $n + 1 \notin s(\mathbb{N}^+)$. Clearly $v \in \mathbb{S}_2^{s(\mathbb{N}^+)}$ and $\Sigma v = u$.

Corollary 3:

Proof. From Proposition 4, Proposition 5 and Theorem 8(6), the sequence t belongs to $\mathbb{S}_3^{s(\mathbb{N}^+)}$. Then, from Theorem 6, the structure $\langle \mathbb{N}, +1, t(\mathbb{N}) \rangle$ has a decidable MSO-theory.

Corollary 2:

[2] here $\binom{j}{i}$ denotes the binomial coefficient

Proof.

1. Let $u_i = 0$ and $u_i(n+1) = \sum_{j=0}^{k_i-1} \binom{j}{k_i} n^j$. Clearly, $\Sigma u_i(n) = n^{k_i}$ and from Theorem 4, $u_i \in \mathbb{S}_2$ since u_i is $\mathbb{N}$-rational. Then, for every $i \in [1, m]$, the sequence $(n^{k_i})_{n\in\mathbb{N}}$ belongs to $\Sigma\mathbb{S}_2$. Applying Corollary 1, the MSO-theory of $\langle \mathbb{N}, +1, \{n^{k_m}\}_{n\geq 0}, \{n^{k_m k_{m-1}}\}_{n\geq 0}, \ldots, \{n^{k_m \cdots k_1}\}_{n\geq 0}\rangle$ is decidable.
2. Let u the sequence defined by $u(0) = 1$ and $u(n+1) = 2^n$ for $n \geq 0$. Clearly, $u \in \mathbb{S}_2$ since it is $\mathbb{N}$-rational (Proposition 1) and $\Sigma v_1 = u$. Applying Corollary 1, the MSO-theory of $\langle \mathbb{N}, +1, v_m(\mathbb{N}), v_{m-1}(\mathbb{N}), \ldots, v_1(\mathbb{N})\rangle$, with $v_1(n) = 2^n$ and $v_{i+1}(n) = 2^{v_i(n)}$ is decidable.

5 Proof of Theorem 8

Fact 1 *Let $k \geq 2$ and $u \in \mathbb{N}^{\mathbb{N}}$. The sequence u belongs to $\Sigma\mathbb{S}_k^{\boldsymbol{N}}$ iff Δu belongs to $\mathbb{S}_k^{\boldsymbol{N}}$.*

This follow easily from point (0) of Theorem 4.

Lemma 7. *Let $k \geq 1$ and $U \in \Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$. Then $\mathsf{E}U \in \Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$.*

Proof. Suppose that $U \in \Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$. We notice that $\Delta\mathsf{E}U = \mathsf{E}\Delta U$. Using Fact 1 and stability of $\mathbb{S}_{k+1}^{\boldsymbol{N}}$ by shift, we obtain that $\mathsf{E}U \in \Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$.

Lemma 8. *Let $k \geq 1$ and $U, V \in \Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$. Then $U + V \in \Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$.*

Lemma 9. *Let $k \geq 2$ and $U, V \in \Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$. Then $U \odot V \in \Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$.*

Proof. Let $U, V \in \Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$. The following identity is well-known:

$$\Delta(U \odot V) = \Delta(U \odot \mathsf{E}V) + U \odot (\Delta V).$$

By Theorem 4, the sequences $U, V, \mathsf{E}V$ all belong to $\mathbb{S}_{k+1}^{\boldsymbol{N}}$, and the righthand side of the above identity must belong to $\mathbb{S}_{k+1}^{\boldsymbol{N}}$. By Fact 1, $U \odot V \in \Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$.

Lemma 10. *Let $k \geq 2$ and $U \in \Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$, $V \in \Sigma\mathbb{S}_k$. Then $U \times V \in \Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$.*

Proof. Let $U = \Sigma u$ and $V = \Sigma v$ for some $u \in \mathbb{S}_{k+1}^{\boldsymbol{N}}$ and $v \in \mathbb{S}_k$. Let us transform the expression $\Delta(U \times V)$ into an expression which does not use the operator Δ any more.

$$\begin{aligned}
\Delta(\Sigma u \times \Sigma v) &= \frac{(\Sigma u \times \Sigma v)(1 - X) - \Sigma u(0) \cdot \Sigma v(0)}{X} \\
&= \frac{\frac{u \times v}{1-X} - \frac{u(0)\cdot v(0)}{1-0}}{X} \\
&= \mathsf{E}(\frac{u \times v}{1 - X}) \\
&= \mathsf{E}\Sigma(u \times v).
\end{aligned}$$

By Theorem 4(3), the final expression obtained belongs to $\mathbb{S}_{k+1}^{\boldsymbol{N}}$, hence $\Sigma u \times \Sigma v$ belongs to $\Sigma\mathbb{S}_{k+1}^{\boldsymbol{N}}$.

Lemma 11. *Let $V \in \Sigma\mathbb{S}_k$, $k \geq 2$, such that $V(0) \geq 1$. Let U be the sequence defined by*

$$U(0) = 1 \text{ and for all } n \geq 0, U(n+1) = \sum_{k=0}^{n} U(k) \cdot V(n-k).$$

Then $U \in \Sigma\mathbb{S}_{k+1}$.

Proof. Let $v \in \mathbb{S}_{k+1}$ such that $V = \Sigma v$. as asserted in 3

$$U = \frac{1}{1 - \frac{Xv}{1-X}}.$$

Let us compute the series ΔU.

$$\Delta U = \Delta(\frac{1}{1 - \frac{Xv}{1-X}}) = \Delta(\frac{1-X}{1-X-Xv}) = \frac{1}{X}[\frac{(1-X)^2}{1-X-Xv} - 1]$$

hence

$$\Delta U = \frac{1}{X}[\frac{(-X + X^2 + Xv)}{1-X-Xv}] \tag{20}$$

Let us compute the series $U \times v - 1$:

$$U \times v - 1 = \frac{(1-X)v}{1-X-Xv} - 1 = \frac{1}{X}[\frac{(-X + X^2 + Xv)}{1-X-Xv}] \tag{21}$$

From Equations (20),(21) we get the identity:

$$\Delta U = U \times v - 1. \tag{22}$$

By the stability properties established in Theorem 4, U belongs to $\mathbb{S}_{k+1}$ and $U \times v$ belongs to $\mathbb{S}_{k+1}$*too*. The hypothesis that $U(0) = 1$ and $V(0) \geq 1$ ensures that $U \times v - 1$ belongs to $\mathbb{S}_{k+1}$. Formula (22) shows that $U \in \Sigma\mathbb{S}_{k+1}$.

Lemma 12. *Let $k_1 \geq 1, k_2 \geq 1, U \in \Sigma\mathbb{S}_{k_1+1}$ and $V \in \Sigma\mathbb{S}^{\boldsymbol{N}}_{k_2+1}$. Then $U \circ V \in \Sigma\mathbb{S}^{\boldsymbol{N}}_{k_1+k_2+1}$.*

Sketch of proof: Let $U = \Sigma u, V = \Sigma v$ for some $u \in \mathbb{S}_{k_1+1}, v \in \mathbb{S}^{\boldsymbol{N}}_{k_2+1}$. Then $U \circ V = \sum_{m=0}^{m=V(n)} u(m)$. Hence

$$(\Delta(U \circ V))(n) = \sum_{m=V(n)+1}^{m=V(n+1)} u_m$$

Let us prove that $\sum_{m=V(n)+1}^{m=V(n+1)} u_m$ belongs to $\mathbb{S}^{\boldsymbol{N}}_{k_1+k_2+1}$.

Let $k = k_1 + k_2 + 1$. Some k-DCPDA computing $U \circ V$ can be constructed along the following lines.

Let us notice that, by Theorem 4, point (6), V belongs also to $\mathbb{S}_{k_2+1}$. By

Lemma 3, there exists $\mathcal{A} \in k_1+1$-DCPDA over pushdown alphabets $A_{k_1+1} \supseteq \{a_{k_1+1}, \mathrm{A}_{k_1+1}\}$ and $A_i \supseteq \{a_i\}$ for $i \in [1, k_1]$, and there exists $\mathcal{B} \in k_2+1$-DCPDA$^{\boldsymbol{N}}$ over pushdown alphabets $B_{k_2+1} \supseteq \{b_{k_2+1}, \mathrm{B}_{k_2+1}\}$ and $B_i \supseteq \{b_i\}$ and there exists $\mathcal{C} \in k_2+1$-DCPDA$^{\boldsymbol{N}}$ over pushdown alphabets $C_{k_2+1} \supseteq \{c_{k_2+1}, \mathrm{C}_{k_2+1}\}$ and $C_i \supseteq \{c_i\}$ for $i \in [1,k]$, $b_1 = c_1$ with sets of states $Q_{\mathcal{A}} \ni q_0, Q_{\mathcal{B}}, Q_{\mathcal{C}}$, chosen in such way as:

$$Q_{\mathcal{B}} \cap Q_{\mathcal{C}} = \{r_0\}$$

$$(q_0 a_k[a_{k-1}[\cdots[a_{k_2+1}{}^n]]\cdots]]q_0) \vdash^*_{\mathcal{A}} (q_0\mathrm{A}_k[\varepsilon]q_0)^{u(n)} \tag{23}$$

$$(r_0 b_{k_2+1}[b_{k_2}[\cdots[b_2[b_1{}^n]]\cdots]]r_0) \vdash^*_{\mathcal{B}} (r_0\mathrm{B}_{k_2+1}[\varepsilon]r_0)^{v(n)} \tag{24}$$

$$(r_0 c_{k_2+1}[c_{k_2}[\cdots[c_2[c_1{}^n]]\cdots]]r_0) \vdash^*_{\mathcal{C}} (r_0\mathrm{C}_{k_2+1}[\varepsilon]r_0)^{V(n)}. \tag{25}$$

Derivation (24) shows the existence of a sequence $H_1, \ldots, H_{v(n)}$ of (k_2+1)-terms fulfilling:

$$H_{v(n+1)} = b_{k_2+1}[b_{k_2}[\cdots[b_2[b_1{}^{n+1}]]\cdots]\Omega_{k_2}], \quad H_1 = \mathrm{B}_{k_2+1}[\Omega_{k_2}],$$

$$(r_0 H_{i+1} r_0) \vdash_{\mathcal{B}} (r_0\mathrm{B}_{k_2+1}[\Omega_{k_2}]r_0)(r_0 H_i[\Omega_{k_2}]r_0).$$

Let $\gamma_n = d_{k_1+1}[\cdots[d_2[d_1{}^n]]\cdots]$. By a construction analogous with that of Proposition 17, we obtain $\mathcal{D} \in k$-DCPDA$^{\boldsymbol{N}}$, over the pushdown alphabets $D_i \supseteq B_i \cup C_i$, for all $i \in [1, k_2+1]$, $D_{k_2+i} \supseteq A_i$, for $i \in [1, k_1+1]$ and D_k contains the new symbols d_k and D_k, and over the set of states $Q_{\mathcal{D}} \supseteq Q_{\mathcal{A}} \times (Q_{\mathcal{B}} \cup Q_{\mathcal{C}})$, making the following rules valid:
argument generation, (G1): for every $n \geq 0$

$$((q_0,r_0)d_k[T_{k-1,k_1+1}[\gamma_n]](q_0,r_0)) \vdash^*_{\mathcal{D}} \prod_{i=1}^{v(n+1)} ((q_0,r_0)T_{k,k_1+1}[H_i \cdot T_{k_1+1,2}[a_1{}^n]](q_0,r_0)),$$

u∘v-computation, (C2): for every $n \geq 0, v(n+1) \geq i \geq 1$

$$(q_0,r_0)T_{k,k_1+2}[H_i \cdot T_{k_1+1,2}[a_1{}^n]](q_0,r_0) \vdash^*_{\mathcal{D}} ((q_0,r_0)\mathrm{D}_k[\varepsilon](q_0,r_0))^{u(i+V(n))}.$$

Combining (G1) and (C2) we finally obtain:

$$\begin{aligned}(q_0,r_0)d_k[T_{k-1,k_1}[\gamma_n]](q_0,r_0) &\vdash^*_{\mathcal{D}} \prod_{i=1}^{v(n+1)} ((q_0,r_0)\mathrm{D}_k[\varepsilon](q_0,r_0))^{u(i+V(n))} \\ &= ((q_0,r_0)\mathrm{D}_k[\varepsilon](q_0,r_0))^{\Delta(U\circ V)(n)}.\end{aligned}$$

Lemma 13. *Let $k \geq 2$. Let $U_1, U_2, \ldots, U_p$ be sequences of integers, $P_1, P_2, \ldots, P_p$ be polynomials in $\Sigma\mathbb{S}^{\boldsymbol{N}}_{k+1}[X_1, X_2, \ldots, X_p]$, $c_1, c_2, \ldots, c_p \in \mathbb{N}$ such that: for all $1 \leq i \leq p$*

$$U_i(n+1) = P_i(n, U_1(n), U_2(n), \ldots, U_p(n)) \text{ and } c_i = U_i(0) \leq U_i(1).$$

Then $U_1 \in \Sigma\mathbb{S}^{\boldsymbol{N}}_{k+1}$.

Proof. Let U_i, P_i, c_i fulfilling the hypothesis of the lemma. Let $a_0(n), a_1(n), \ldots, a_q(n)$ be a sequence enumerating all the coefficients of the polynomials $P_1, P_2, \ldots, P_p$.

There exists polynomials $Q_i \in \mathbb{N}[X_0, \ldots, X_{q+p}]$, such that for all $i \in [1, p]$:

$$P_i(n, U_1(n), U_2(n), \ldots, U_p(n)) = Q_i(a_0(n), \ldots, a_q(n), U_1(n), \ldots, U_p(n)).$$

The Euler-Mac-Laurin formula applied to polynomials Q_i expresses the difference

$$Q_i(X_0, \ldots, X_{q+p}) - Q_i(Y_0, \ldots, Y_{q+p})$$

under the form:

$$\sum_{\bar{k}} \frac{1}{\bar{k}!} \frac{\partial^{\bar{k}} Q_i}{(\partial X_0)^{k_0} \ldots (\partial X_{q+p})^{k_{q+p}}}(Y_0, \ldots, Y_{q+p}) \cdot (X_0 - Y_0)^{k_0} \cdots (X_{q+p} - Y_{q+p})^{k_{q+p}}, \tag{26}$$

where $\bar{k} = (k_1, k_2, \ldots, k_{q+p})$ varies over all the $(q+p)$-tuples with sum $k_1 + k_2 + \ldots + k_{q+p}$ smaller of equal to the degree Q_i. For every monomial $M = X_0^{d_0} X_1^{d_1} \cdots X_{q+p}^{d_{q+p}}$ the partial derivative

$$\frac{1}{\bar{k}!} \frac{\partial^{\bar{k}} M}{(\partial X_0)^{k_0} \ldots (\partial X_{q+p})^{k_{q+p}}}(Y_0, \ldots, Y_{q+p}),$$

is equal to

$$\binom{d_0}{k_0}\binom{d_1}{k_1} \cdots \binom{d_{q+p}}{k_{q+p}} \cdot Y_0^{d_0 - k_0} Y_1^{d_1 - k_1} \cdots Y_{q+p}^{d_{q+p} - k_{q+p}}. \tag{27}$$

Every partial derivative

$$R_{i,\bar{k}} = \frac{\partial^{\bar{k}} Q_i}{(\partial X_0)^{k_0} \ldots (\partial X_{q+p})^{k_{q+p}}}(Y_0, \ldots, Y_{q+p})$$

is a linear combination, with coefficients in $\mathbb{N}$, of monomial of the form (27), hence it has only non-negative integer coefficients:

$$R_{i,\bar{k}} \in \mathbb{N}[Y_0, \ldots, Y_{q+p}].$$

Let us apply the following substitution to the indeterminates $X_0, \ldots, X_{q+p}, Y_0, \ldots, Y_{q+p}$,

$$X_j \leftarrow a_j(n+1) \text{ pour } 0 \le j \le q; \quad X_{q+\ell} \leftarrow U_\ell(n+1) \text{ pour } 0 \le \ell \le p,$$

$$Y_j \leftarrow a_j(n) \text{ pour } 0 \le j \le q; \quad Y_{q+\ell} \leftarrow U_\ell(n) \text{ pour } 0 \le \ell \le p.$$

We obtain: $(\Delta U_i)(n+1) =$

$$\sum_{\bar{k}} R_{i,\bar{k}}(a_0(n+1), \ldots, a_q(n+1), U_1(n), \ldots, U_p(n)) \cdot (\Delta \bar{a}(n))^{\bar{k}} \cdot (\Delta \bar{U}(n))^{\bar{k}} \tag{28}$$

where the expression $(\Delta \bar{a})^{\bar{k}}(n)$ means:

$$(\Delta a_0)^{k_0}(n) \cdots (\Delta a_q)^{k_q}(n)$$

and the expression $(\Delta\bar{U})^{\bar{k}}(n)$ means:

$$(\Delta U_1)^{k_{q+1}}(n) \cdots (\Delta U_p)^{k_{q+p}}(n).$$

By the closure properties established in Theorem 4, every sequence $R_{i,\bar{k}}(a_0(n+1), \ldots, a_q(n+1), U_1(n), \ldots, U_p(n))$ belongs to $\mathbb{S}_{k+1}^{\boldsymbol{N}}$.